VYSOKÁ ŠKOLA BÁŇSKÁ – TECHNICKÁ UNIVERZITA OSTRAVA
EKONOMICKÁ FAKULTA

KATEDRA SYSTÉMOVÉHO INŽENÝRSTVÍ A INFORMATIKY

Využití metod operačního výzkumu pro optimalizaci trasy při zásobování

Usage of Operations Research Methods in Delivery Route Optimization

Student: Jan Hastík

Vedoucí bakalářské práce: Ing. Eva Moravcová, CSc.

Ostrava 2008

„Místopřísežně prohlašuji, že jsem celou práci včetně všech příloh vypracoval samostatně“.

V Ostravě dne 24. dubna 2008

……...…………………….

Jan Hastík

OBSAH

ÚVOD **2**

1 OPERAČNÍ VÝZKUM **4**

1.1 PODSTATA A METODY OPERAČNÍHO VÝZKUMU 4

2 PŘÍSTUPY K ŘEŠENÍ PROBLÉMU OBCHODNÍHO CESTUJÍCÍHO **6**

2.1 DEFINICE PROBLÉMU OBCHODNÍHO CESTUJÍCÍHO 6

2.1.1 NP-úplný problém 6

2.1.2 Heuristické algoritmy 7

2.1.3 Matematické vyjádření úlohy obchodního cestujícího 7

2.2 POJMY Z TEORIE GRAFŮ 8

2.3 METODY K ŘEŠENÍ PROBLÉMU OBCHODNÍHO CESTUJÍCÍHO 10

2.4 HAMILTONOVA KRUŽNICE 10

2.4.1 Podmínky existence Hamiltonovy kružnice 10

2.4.2 Heuristický algoritmus určení minimální HK 11

2.5 KIMOVA METODA 12

2.5.1 Algoritmus Kimovy metody 12

2.6 ROZHODOVACÍ STROMY 13

2.6.1 Algoritmus rozhodovacího stromu 13

2.7 METODA PENALIZACÍ 15

2.7.1 Algoritmus metody penalizací 15

2.8 METODA VĚTVÍ A MEZÍ 16

2.8.1 Principy a pravidla metody větví a mezí 17

2.8.2 Obecný algoritmus metody větví a mezí 18

2.9 ZHODNOCENÍ A VÝBĚR METODY PRO ŘEŠENÍ PROBLÉMU OBCHODNÍHO CESTUJÍCÍHO 19

3 POPIS PROBLÉMU **20**

3.1 CHARAKTERISTIKA FIRMY 20

3.1.1 Způsob zásobování 20

3.1.2 Dostupný software 20

3.2 POŽADAVKY NA APLIKACI 22

3.3 CHARAKTERISTIKA SOUČASNÉHO STAVU ... 22
4 APLIKACE VYBRANÉ METODY ... 27
4.1 APLIKACE METODY ROZHODOVACÍHO STROMU ... 27
4.1.2 Návrh k použití metody RS pro větší počet uzlů ... 28
4.2 POPIS VYTVOŘENÉ APLIKACE ... 29
5 HODNOCENÍ PŘÍNOSU ŘEŠENÍ ... 33
5.1 DÉLKA PŮVODNÍ TRASY ... 33
5.2 DÉLKA TRASY PO OPTIMALIZACI, INTERPRETACE ŘEŠENÍ ... 33
ZÁVĚR ... 36
SEZNAM POUŽITÉ LITERATURY ... 38

Úvod

V dnešní moderní době je vyvíjen stále větší tlak na snížení výrobních nákladů, a proto se firmy snaží co nejvíce optimalizovat své výrobní procesy. Chce-li být firma úspěšná a konkurenceschopná, musí se snažit o optimální alokaci svých zdrojů, minimalizaci spotřeby vstupních nákladů při současné maximalizaci výstupu, musí usilovat o co nejvhodnější způsob zásobování atp. Není-li firma schopna dostatečně řídit své operace a procesy tak, aby dokázala obstát v silné konkurenci, potom dochází ke značným ztrátám, částečnému omezení výroby nebo dokonce k jejímu zániku.

Operační výzkum představuje nástroj pro optimální řešení daného rozhodovacího problému. Původně vznikl jako samostatný vědní obor především pro praktické potřeby ve vojenské oblasti. Samotný termín operační výzkum byl pravděpodobně poprvé použit v roce 1938 ve Velké Británii [9]. Postupem času se začal operační výzkum využívat i mimo oblast ryze vojenských problémů a začínal nacházet uplatnění především v poválečném průmyslu. Vývoj metod operačního výzkumu neustále probíhá, dochází k zániku zastaralých metod a vzniku metod nových [5].

Převážná většina prakticky využívaných metod operačního výzkumu je v praxi spjata s využitím dokonalé výpočetní techniky. Z tohoto důvodu vyžadují tyto metody algoritmizaci, aby mohly být programově zrealizovány [9].

Motto Ústavu pro operační výzkum na Univerzitě George Washingtona výstižně charakterizuje podstatu celého operačního výzkumu: „Operační výzkum: lépe, chytřeji, rychleji [12] !“

Zásobování představuje důležitou a neodmyslitelnou součást firemních procesů, a proto této oblasti věnují firmy velkou pozornost. Jde o činnost, při které (si) podnik zajišťuje potřebné suroviny (zboží) v požadovaném množství, kvalitě a ve stanovené době.

Firma Kamahaj zabývající se výrobou domácích cukrářských výrobků si stanovila požadavek možnosti zkrácení stávající trasy, po které provádí zásobování svými produkty. Protože si zásobování firma provádí sama, představuje tato problematika druhořadou oblast jejího zájmu. Nebyla proto nikdy věnována dostatečná pozornost na zefektivnění tohoto procesu. K obsluze míst sice firma volila logicky nejkratší možné spojení, ale nikdy nebyla provedena optimalizace. Tím, že dojde k možnému zkrácení délky trasy, bude firmě umožněna efektivnější obsluha míst a dojde k časové i nákladové úspoře.

Ve své bakalářské práci se zabývám možnými přístupy operačního výzkumu k problému obchodního cestujícího. S tímto problémem, známým též jako okružní problém, se setkávají firmy a podniky, které provádějí zásobování, resp. rozvoz a svoz výrobků a materiálů. Přitom je žádoucí, aby dané místo navštívily právě jednou a urazily přitom nejkratší možnou vzdálenost.

Cílem této práce je seznámení s možnými přístupy k řešení problému obchodního cestujícího a navržení jednoduché aplikace založené na použití zvolené metody, popř. kombinace metod k POC. Aplikace umožní firmě zefektivnit stávající trasu, po které provádí zásobování do jednotlivých provozoven a přispěje tak k částečnému snížení nákladů vynaložených na tuto činnost. Program navíc poslouží firmě i při případných budoucích změnách v počtu či umístění míst, do kterých směřuje zásobování. Opětovně vyhledá a přepočítá délky nových optimálních tras.

Aby byl cíl práce náležitě splněn, je třeba postupovat v těchto krocích:

1) vymezení teoretických přístupů k řešení problému obchodního cestujícího na základě operačního výzkumu,
2) srovnání jednotlivých přístupů vymezených v bodě 1), určení jejich silných a slabých stránek důležitých pro konečné rozhodnutí spočívající ve výběru vhodné metody k aplikaci,
3) výběr metody k řešení daného problému,
4) analýza současného stavu, z toho vyplývající požadavky na aplikaci,
5) návrh aplikace využívající zvolenou metodu v bodě 3),
6) zhodnocení výsledku dosaženého optimalizací původní trasy.

Takto provedená dekompozice hlavního cíle na podcíle umožňuje problém lépe pochopit, vytyčit silné a slabé stránky jednotlivých metod a v konečném důsledku vykonat správné rozhodnutí pro výběr metody.

1 Operační výzkum

1.1 Podstata a metody operačního výzkumu

Operační výzkum představuje vědní disciplínu vyvíjející a aplikující exaktní postupy pro řešení ekonomického, organizačního, technického či vojensko-strategického problému. Problém je řešen zpravidla týmem specialistů různého odborného zaměření. Při řešení problému je zkoumaný objekt považován za složitý systém, k jehož zkoumání a řešení se využívá znalostí různých vědních disciplín. Základním nástrojem operačního výzkumu je matematické modelování s využitím výpočetní techniky [5]. Matematický model je zformalizovaný ekonomický model rozhodovacího problému, který je možný řešit standardními postupy [1].

Výsledkem řešení problému metodou operačního výzkumu je:

- vypracování návrhu řešení závažného problému jako podkladu pro kvalifikované rozhodnutí,
- vypracování pracovní metody, jejíž opakované použití má podstatně zlepšit úroveň rozhodování v řízení na daném úseku [5].

Operační výzkum se při řešení problémů reálného objektu nesnaží tento objekt rozčlenit na dílčí složky a ty pak izolovaně zkoumat. Postup OV vychází z předpokladu, že objekt pojímaný jako složitý systém je něco více než jen souhrn prvků, ze kterých se daný systém skládá. Snaha OV při řešení problému je proto vymezit systém, pro který problém existuje a pak vymezit jeho okolí. Vymezený systém zkoumá jako celek. Uvedený postup patří k podstatným rysům operačního výzkumu a označuje se jako systémový přístup k řešení problému [5].

Kromě systémového přístupu se operační výzkum vyznačuje:

- týmovou prací specialistů různých oborů a zaměření,
- použitím speciálních metod,
- použitím modelové techniky opírající se o matematicko-statistický aparát.

Pro řešení nejrůznějších problémů praxe byly postupem času vyvinuty speciální metody, které řeší typické situace. Operační výzkum se stále vyvíjí, dochází ke vzniku nových a zániku starých a neosvědčených metod. Neexistuje proto jednotné třídění speciálních metod OV. Z hlediska těch metod, které našly při řešení ekonomických a technicko-ekonomických problémů širší použití, můžeme speciální metody operačního výzkumu rozdělit do dvou hlavních skupin [5]:

1) ***Metody pro analýzu struktury a chování systému a jeho okolí***
   - metody síťové analýzy,
   - metody strukturní analýzy,
   - metody optimální alokace omezených zdrojů,
   - metody analýzy okolí,
   - metody pro řešení problémů soutěže (teorie her),
   - simulační techniky.

2) ***Modely hlavních subsystémů***
   - zásobovací modely,
   - modely teorie obnovy,
   - modely teorie hromadné obsluhy.

Metodami operačního výzkumu lze řešit téměř jakýkoliv problém na úsecích ekonomiky podniku, organizace, odvětví, národního hospodářství. K základním oblastem využití operačního výzkumu patří:

- plánování na různých úrovních řízení,
- řízení výroby,
- řízení skladovací a zásobovací činnosti,
- řízení výzkumu a vývoje,
- řešení personálních, finančních otázek a některých otázek účetnictví, statistiky apod. [5].

2 Přístupy k řešení problému obchodního cestujícího

2.1 Definice problému obchodního cestujícího

Problém obchodního cestujícího, nebo-li okružní dopravní problém, je obtížný diskrétní optimalizační problém, který řeší nalezení minimální (tedy nejkratší) uzavřené trasy, která by začínala a končila v daném uzlu V_0 a současně procházela každým uzlem právě jednou [6], [10]. Jinými slovy hledáme takový Hamiltonův cyklus [1], který minimalizuje součet ohodnocení jeho hran [7].

Název této úlohy vznikl z původní interpretace, kdy obchodní cestující má za úkol projít všechna stanovená města a vrátit se do výchozího města tak, aby přitom urazil nejkratší vzdálenost [7]. Úloha má řadu praktických uplatnění především tam, kde se jedná o pravidelný rozvoz či svoz produktů (zásobování prodejen, výběr poštovních schránek, svoz odpadu atd.) [6]. Požadavek minimální navrhované trasy vede ke zkrácení doby obsluhy, nižší spotřebě pohonných hmot a tím i k nižším nákladům.

2.1.1 NP-úplný problém

Úloha obchodního cestujícího patří mezi NP-úplné problémy[2] (nedeterministické polynomiální problémy), které se vyznačují obtížnou řešitelností při rostoucím počtu uzlů. Problém spočívá v nalezení přesného řešení v reálném čase s využitím časově efektivního algoritmu, který nebyl dosud pro tuto úlohu nalezen.

V praxi při hledání řešení úlohy obchodního cestujícího používáme heuristické metody. Při využití těchto metod sice získáme řešení v přípustném čase, ale to je pouze suboptimální. Suboptimální řešení je často vyhovující, ale jen málokdy nejlepší možné, navíc heuristické metody neumožňují odhadnout, jak moc se nalezené řešení liší od optimálního [2], [6], [11].

[1] O Hamiltonovu cyklu bude dále pojednáno v kapitole 2.4.

[2] V současnosti je známo kolem 1 000 NP-úplných problémů a jejich počet stále vzrůstá. Pro žádný z těchto NP-úplných problémů nikdo dosud nenašel polynomiální algoritmus. Na druhou stranu nikdo nedokázal neexistenci časově polynomiálního algoritmu pro některý z nich [2].

2.1.2 Heuristické algoritmy

Většina heuristických metod je založena na dvou různých postupech:

1) Je vytvořeno výchozí, většinou zcela libovolné přípustné řešení, které je dále postupně optimalizováno při dodržení podmínek přípustnosti. Proces zlepšování řešení končí při dosažení přípustného řešení, které už nelze žádnými změnami dále zlepšit. Tento postup řešení se nazývá ***primární heuristika***.

2) Je vytvořeno nepřípustné výchozí řešení, které má hodnotu účelové funkce lepší než optimální řešení. Toto výchozí řešení je dále upravováno tak, aby se postupně odstranilo narušení podmínek přípustnosti při současném minimálním zhoršení hodnoty účelové funkce v jednotlivých krocích. Řešení probíhá tak dlouho, dokud nezískáme nějaké přípustné řešení nebo pokud už nelze lepšího řešení dosáhnout. Jedná se o tzv. ***duální heuristiku*** [6].

2.1.3 Matematické vyjádření úlohy obchodního cestujícího

Mezi nejpropracovanější a nejpoužívanější disciplíny operačního výzkumu patří tzv. ***matematické programování*** [3]. Představuje skupinu metod operačního výzkumu, v nichž subjekt rozhodování musí optimálně alokovat omezené zdroje [5].

Lineární programování je soubor metod matematického programování využívající nástroje lineární algebry [1].

Při praktické aplikaci lineárního programování je problém rozdělen do několika – zpravidla čtyř fází:

- ***Formulace ekonomického modelu*** – odráží zkoumanou problematiku pomocí slovního a numerického popisu problému.
- ***Formulace matematického modelu*** – převedený ekonomický problém, který je řešitelný matematickými postupy.
- ***Výpočet matematického modelu*** užitím vhodné metody lineárního programování.
- ***Ekonomická nebo věcná interpretace řešení*** – zpětný převod výsledku matematického řešení do ekonomické řeči [5].

Matematický model problému obchodního cestujícího lze zapsat:

minimalizace účelové funkce $z = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij} x_{ij} \longrightarrow \min,$ (2.1)

za podmínek

$$\sum_{i=1}^{n} x_{ij} = 1, \qquad j = 1, 2, \dots, n, \qquad (2.2)$$

$$\sum_{j=1}^{n} x_{ij} = 1, \qquad i = 1, 2, \dots, n, \qquad (2.3)$$

$$\delta_i - \delta_j + (n-1)x_{ij} \le n-2, \qquad i = 2, 3, \dots, n; j = 1, 2, \dots, n-1; i \neq j, \qquad (2.4)$$

$$x_{ij} = \{0;1\}, \qquad i, j = 1, 2, \dots, n. \qquad (2.5)$$

Podmínka (2.2) zajišťuje, že obsluhující komplet vstoupí do každého uzlu právě jednou. Podmínka (2.3) zajišťuje, že komplet z každého uzlu právě jednou vystoupí. Podmínka (2.4) zabraňuje rozpadu trasy na několik nesouvislých částí ($\delta_i - \delta_j$ jsou libovolná reálná čísla). Poslední podmínka (2.5) říká, že řešením úlohy je vektor nabývající pouze hodnot 0 a 1. Proměnnou x_{ij} lze tedy charakterizovat jako diskrétní bivalentní (dvouhodnotovou) proměnnou. Je-li $x_{ij} = 0$, tj. nabývá-li proměnná hodnoty nula, úsek do Hamiltonovy kružnice zařazen není. Je-li $x_{ij} = 1$, úsek do HK použijeme. Množina přípustných řešení je určena množinou všech vektorů vyhovující podmínkám (2.2), (2.3), (2.4). Nenulový prvek vektoru udává použití úseku a naopak [1].

2.2 Pojmy z teorie grafů

Matematická formulace úlohy obchodního cestujícího vychází z pojmosloví ***teorie grafů***. Teorie grafů se zabývá studiem vlastností útvarů, které nazývá grafy. Graf je útvar, který lze znázornit v rovině pomocí bodů a spojnic mezi nimi. Body se označují jako uzly, spojnice mezi jednotlivými uzly jsou hrany grafu.

Přehled pojmů z teorie grafů použitých v práci

Úplný graf je takový graf, kdy každý uzel je spojen hranami se všemi ostatními uzly. Vynecháme-li z grafu některé uzly a příslušné hrany, které jsou incidentní s vynechanými uzly, nazýváme takto vzniklý graf *podgrafem* původního grafu. *Sousední uzly* jsou nazývány takové uzly u_i a u_j, pro které platí h_{ij}.

Počet hran incidentních s daným uzlem nazýváme *stupněm uzlu*. Jestliže je stupeň uzlu stejný pro všechny uzly daného grafu, nazýváme jej *pravidelným grafem*.

Sled mezi uzly x_0 *a* x_n je konečná posloupnost uzlů a hran daného grafu, která má tvar $x_0, x_0x_1, x_1, x_1x_2, x_2, \ldots, x_{n-1}, x_{n-1}x_n, x_n$.

Otevřený sled, jehož všechny uzly jsou různé (každý uzel se vyskytne nejvýše jedenkrát) se nazývá *cesta.*

Cesta, která začíná a končí v jednom uzlu, se nazývá v neorientovaném grafu *kružnice*, v orientovaném *cyklus*.

Pojmy související s podmínkami existence Hamiltonovy kružnice

Souvislý graf je takový graf, v němž platí, že pro každé dva vrcholy existuje alespoň jedna cesta.

Jestliže je počet uzlů *n* konečný, hovoříme o *konečném grafu*, v opačném případě jde o *graf nekonečný*.

Obyčejný graf budeme nazývat takový graf $G = (U, H)$, kde U je množina uzlů a H je množina hran a každé dva vrcholy spojuje nejvýše jedna hrana.

Komponent grafu je maximální souvislý podgraf grafu. Má-li graf jeden komponent, potom je graf souvislý (izolovaný vrchol považujeme za samostatný komponent grafu).

Most je hrana, jejíž odstraněním z grafu se počet komponent grafu zvýší o jeden.

Artikulace je vrchol, jehož odstraněním z grafu včetně všech hran incidujících s tímto vrcholem se zvýší počet komponent grafu alespoň o jeden [7], [8].

2.3 Metody k řešení problému obchodního cestujícího

Existuje několik možných přístupů k řešení problému obchodního cestujícího. Jednotlivé přístupy, které budou dále podrobněji rozebrány, využívají těchto metod:

- metoda využívající poznatků teorie grafů (Hamiltonova kružnice, eulerovský tah…),
- metoda větveného rozhodování (rozhodovací stromy)[3],
- metoda lineárního programování.

Některé přístupy k řešení POC nelze jednoznačně zařadit do jediné z uvedených metod řešení. Při konkrétní aplikaci dochází u přístupů k prolínání více metod.

2.4 Hamiltonova kružnice

Jednou z metod využívajících poznatků teorie grafů pro řešení problému obchodního cestujícího je přístup, který spočívá v nalezení tzv. minimální Hamiltonovy kružnice. Hamiltonova kružnice[4], resp. hamiltonovský cyklus se nazývá cyklus, který prochází každým uzlem grafu právě jednou a ve kterém existuje mezi výchozím a koncovým uzlem platná cesta. Graf, který obsahuje Hamiltonovu kružnici se nazývá hamiltonovský [6], [8].

2.4.1 Podmínky existence Hamiltonovy kružnice

Pro existenci uzavřené cesty, která by procházela každým uzlem sítě právě jednou, jsou známy postačující podmínky a také nutné podmínky existence. Podmínky, které by byly nutné a zároveň postačující nejsou doposud známy[5] [8]. Nelze proto rozhodnout, která úloha je řešitelná a která není s použitím jednoduchého kritéria.

---

[3] Už samotný název rozhodovacího stromu napovídá, že i tato metoda využívá poznatků teorie grafů. Budeme ji ale chápat jako samostatnou metodu, která je založena na postupném rozvíjení jednotlivých variant řešení.
[4] Hamiltonova kružnice je pojmenována po irském matematikovi a fyzikovi W. R. Hamiltonovi, který vyřešil problém o tom, zda lze po hranách do roviny rozvinutého pravidelného dvanáctistěnu projít právě jedenkrát všemi jeho vrcholy a vrátit se zpět do výchozího bodu [6].
[5] Z uvedeného plyne složitost určení minimální Hamiltonovy kružnice.

Nutné podmínky existence Hamiltonovy kružnice jsou:

- Souvislý, obyčejný a konečný graf, který obsahuje alespoň tři uzly.
- Graf nesmí obsahovat mosty a artikulace.

Splnění nutných podmínek není postačující pro existenci Hamiltonovy kružnice.

Postačující podmínky existence Hamiltonovy kružnice jsou:

- Součet stupňů každých dvou nesousedních uzlů grafu *G = (V, X)* je větší nebo roven počtu uzlů v síti *n*, přitom musí platit: $|V| = n \geq 3$.
- Stupeň každého uzlu grafu je alespoň *n / 2*, kde *n* je počet uzlů sítě.
- Jestliže pro každé přirozené číslo $1 \leq k \leq \frac{1}{2}n$ je počet uzlů, jejichž stupeň nepřevyšuje *k*, menší než *k* [8].

Jestliže nejsou tyto postačující podmínky splněny, může přesto Hamiltonova kružnice v grafu existovat[6].

2.4.2 Heuristický algoritmus určení minimální Hamiltonovy kružnice

1) Konstrukci HK začneme v libovolném bodu uzlu V_0. Ke zvolenému uzlu V_0 vybereme s ním incidentní hranu s minimálním ohodnocením. Existuje-li více takových hran, vybereme libovolnou z nich.
2) K uzlu, ve kterém se při konstrukci aktuálně nacházíme, vybereme takovou s ním incidentní hranu, jejíž ohodnocení je minimální a jejíž koncový uzel ještě nebyl do HK vybrán. Existuje-li více takových hran, vybereme libovolnou z nich.
3) Postup kroku 2) opakujeme tak dlouho, dokud je možné vybrat alespoň jednu další hranu požadované vlastnosti. Jestliže není možné krok 2) dále opakovat, vybereme do HK tu hranu, která spojuje naposled vybraný uzel určený výběrem poslední hrany s uzlem V_0. Při výběru poslední hrany nebere v úvahu její ohodnocení [6].

[6] Hamiltonovu kružnici je možné nalézt pomocí metody větví a mezí (bude dále podrobněji rozebrána). Tato metoda je výhodná, pokud chceme zpracovávat menší grafy. Přednost této metody spočívá v tom, že pokud existuje v grafu Hamiltonova kružnice, tak bude vždy nalezena. U rozsáhlých grafů je však účelné použití některé z heuristických metod, které jsou rychlé, ale nemusí nalézt kružnici, i když existuje [7].

Tento algoritmus se také nazývá ***metoda nejbližšího dosud nenavštíveného vrcholu***. Nevýhodou této metody je, že ne vždy vyhledá optimální řešení. To je způsobeno stavbou algoritmu, kdy volba např. první (nejkratší) hrany má vliv na výběr dalších hran a tím i na konečnou celkovou délku trasy. Jednoduchá aplikace a nenáročnost výpočtu jsou významnými činiteli, proč tuto metodu uplatnit i při řešení úloh s větším počtem uzlů.

2.5 Kimova metoda

Další metodou, která využívá poznatků teorie grafů je Kimova metoda. Tato heuristická metoda vyhledává rovněž suboptimální řešení POC.

Dříve, než přistoupíme k samotnému algoritmu řešení Kimovy metody, je třeba definovat další dva pojmy z teorie grafů, se kterými bude tato metoda operovat. Jde o eulerovskou síť a eulerovský tah.

Eulerovská síť je taková síť (graf), jejíž každý vrchol má sudý stupeň. *Eulerovský tah* je takový tah[7], který obsahuje všechny hrany grafu a každou hranu obsahuje právě jednou. Eulerovské tahy a sledy mohou být orientované nebo neorientované, uzavřené (začíná a končí ve stejném vrcholu) nebo otevřené [7], [8].

2.5.1 Algoritmus Kimovy metody

1) Danou síť grafu *G* doplníme na úplnou síť *G´* tak, že ke každému doplňovanému, dosud v síti neexistujícímu, úseku přiřadíme vzdálenost trasy, která odpovídá příslušné dvojici uzlů původní sítě, mezi které je úsek vkládán.
2) V síti upravené podle prvního kroku algoritmu najdeme minimální kostru *K*. Úseky kostry *K* zdvojíme a tím získáme eulerovskou siť.
3) U kostry K najdeme některý eulerovský tah.
4) Nalezený eulerovský tah v původní síti *G´* zkracujeme tak, že pokud přes nějakou skupinu uzlů *Vi,…, Vj*, kde $i < j$ prochází tento tah více než jednou a existuje-li úsek (V_{i-1}, V_{j+1}), zkrátíme část tahu $(V_{i-1}, V_i, \ldots, V_j, V_{j+1})$ tak, že vynecháme úsek $(V_i, \ldots, V_j)$. Tím dochází ke zkrácení hodnoty dosavadního řešení, jež označíme např. *Z*, o hodnotu rovnou: $Z = \sum_{k=i}^{j+1} d(V_{k-1}, V_k) - d(V_{i-1}, V_{j+1})$.
5) Krok čtvrtý opakujeme tak dlouho, dokud je $Z > 0$ [6].

[7] Tah je taková posloupnost, u které se mohou opakovat vrcholy, ale ne hrany.

Protože se jedná o heuristický algoritmus, je nalezené řešení opět suboptimální. Při aplikaci této metody v konkrétním řešeném problému je možné první krok algoritmu vynechat, protože mám danou matici vzdáleností, která udává vzdálenosti z každého uzlu do každého z dalších uzlů. Použití této metody je vhodné u problému, který si můžeme graficky znázornit. U počítačového zpracovaní by použití této metody vyžadovalo složité vyhledávání uzlů zastoupených v minimální kostře více než jedenkrát a prověřování, který z uzlů v posloupnosti uzlů vynechat. Při minimalizační úloze je požadavek vynechání toho úseku, který má větší vzdálenost s porovnávaným úsekem.

2.6 Rozhodovací stromy

Další metodou pro řešení problému obchodního cestujícího je metoda rozhodovacího stromu založená na postupném prohledávání všech možných tras (okruhů). Na rozdíl od výše uvedených způsobů řešení nepatří tato metoda mezi heuristické. Proto je její použití v případě rozsáhlých sítí velmi časově náročné a je určena zejména pro výpočet menších sítí.

Rozhodovací strom představuje účelové a zjednodušené grafické zobrazení struktury rozhodovacího procesu. Zobrazení tohoto procesu pak představuje taková posloupnost uzlů a hran, která vykazuje splnění podmínek optimality.

Tato metoda nachází uplatnění všude tam, kde hledáme optimální variantu rozhodnutí z konečného počtu možností. Při konstrukci stromu zvažujeme všechny možnosti pro každou situaci a nakonec vybereme dle určitého kritéria optimální posloupnost rozhodování. Řešení úlohy obchodního cestujícího je tak pouze jednou z možností aplikace této metody [6].

2.6.1 Algoritmus rozhodovacího stromu

- spočívá v transformaci původní sítě grafu do grafu typu strom.

1) Označíme výchozí uzel V_0, který bude tvořit kořen budoucího rozhodovacího stromu. Z uzlu V_0 zvolíme úsek, který je s tímto uzlem incidentní, do libovolného uzlu. Do kořenu žádný úsek nevstupuje, kromě nakonec určené poslední hrany, která uzavírá kružnici.

2) Po průchodu k úseky je obchodní cestující ve stavu vyjádřeném uzlem k-té úrovně. Z každého uzlu stromu vychází tolik větví, kolik možností může cestující zvolit pro další pokračování v trase, je-li v situaci tímto uzlem vyjádřené.
3) Jestliže se obchodní cestující nachází v uzlu k-té úrovně, může dále pokračovat do dalšího uzlu s výjimkou uzlu, kde se právě nachází a s výjimkou těch uzlů, které již při své cestě navštívil. To se nevztahuje na výchozí uzel V_0, který je i zároveň posledním uzlem konstruovaného stromu.
4) Přípustným řešením je v transformovaném grafu trasa, která začíná a končí ve výchozím uzlu V_0. Tato trasa je složena právě z tolika úseků, kolik uzlů se v dané síti nachází. Jedná se tedy o 1-strom[8]. Optimální řešení je hamiltonovská trasa s minimální délkou, tj. minimální 1-strom [6].

Metodu lze také snadno aplikovat na orientovaných sítích. Orientace pak přikazuje směr jejího průchodu a tedy i výběr úseku, který bude tvořit strom.

Hamiltonova kružnice nemusí nutně existovat ve všech sítích. Takovou sítí je jistě neorientovaná síť, která obsahuje uzly stupně jedna. U orientovaných grafů je touto sítí ta, která obsahuje uzly, jehož výstupní stupeň je nula. Jestliže se problém neexistence HK v síti vyskytne, můžeme i přesto úlohu obchodního cestujícího vyřešit, a to za předpokladu zmírnění podmínek zadání úlohy. Nahradíme-li striktní požadavek problému obchodního cestujícího na navštívení každého uzlu právě jednou na úlohu, ve které obchodní cestující navštíví každý uzel alespoň jednou, dostaneme tzv. ***modifikovaný problém obchodního cestujícího***. Tato úloha má již řešení vždy, ale jeho hledání může být značně komplikované i proto, že efektivní algoritmus pro modifikaci úlohy není znám. Možností řešení je například použití rozhodovacího stromu s jinými kritérii jeho konstrukce, než jaké byly uvedeny pro původní formulaci úlohy[9] [6].

Velkou předností této metody je, že poskytuje optimální řešení. Naproti tomu je použití rozhodovacího stromu nevhodné pro řešení rozsáhlých sítí, protože časová a výpočetní náročnost značně omezuje její uplatnění. Protože problém, na který budu metodu aplikovat, je nenáročný na počet uzlů, metodu je vhodné použít.

---

[8] Pojem1-strom bude podrobněji rozebrán v následující kapitole.
[9] Tento postup je však často velmi zdlouhavý a náročný.

2.7 Metoda penalizací

Metoda penalizací je založena na pojmu tzv. ***1-stromu***. Podgraf souvislého grafu G se nazývá 1-strom, jestliže je souvislý, obsahuje všechny uzly grafu G a má n uzlů a n hran. Vznikne z kostry přidáním jedné libovolné hrany. Máme-li hranově ohodnocený graf, pak minimální 1-strom je 1-strom s nejmenším součtem ohodnocení hran. Minimální 1-strom získáme tak, že k minimální kostře přidáme takovou hranu, která neleží v minimální kostře a která má nejmenší ohodnocení. 1-strom lze sestrojit použitím algoritmu pro nalezení minimální kostry.

Hamiltonova kružnice je tak podgrafem daného grafu G, který je 1-stromem a zároveň všechny jeho uzly mají stupeň 2. Součet ohodnocení hran minimálního 1-stromu je dolním odhadem hodnoty řešení úlohy obchodního cestujícího. Jestliže sestrojíme minimální 1-strom, který má všechny uzly stupně 2, pak jsme našli minimální HK. Pokud má některý z uzlů 1-stromu jiný stupeň než je 2, pak získaný graf není HK a aplikujeme metodu penalizací.

Metoda je založena na přiřazení reálného čísla p_i jednotlivým uzlům grafu. Číslo se nazývá penále. Jestliže stupeň daného uzlu minimálního 1-stromu je větší než 2, pak $p_i > 0$. Jestliže je stupeň uzlu menší než 2, pak je penále záporné. Změníme-li ohodnocení jednotlivých hran grafu z hodnoty k_{ij} na $k_{ij} + p_i + p_j$, pak se nemění řešení úlohy obchodního cestujícího, ale mohou se změnit minimální 1-stromy. V nově ohodnoceném grafu pak opět hledáme minimální 1-strom. Protože se penalizací nemění požadované řešení, snažíme se změnami penále ovlivňovat tvar 1-stromu tak, abychom získali kružnici. Délka této nalezené Hamiltonovy kružnice bude nejkratší ze všech možných HK, neboť délky všech HK se změnily o stejnou hodnotu [6], [7].

2.7.1 Algoritmus metody penalizací

1) Inicializace: Všem uzlům přiřadíme hodnoty $v_i = 0, i = 1, 2, \ldots, n$.
2) Nalezení minimálního 1-stromu: V grafu s ohodnocením hran $k_{ij} + v_i + v_j$ určíme minimální 1-strom T.
3) Test ukončení: Je-li T Hamiltonovou kružnicí, výpočet končí optimálním řešením úlohy obchodního cestujícího. Pokud T netvoří HK, přejdeme k následujícímu bodu.

4) Změna inicializace: Hodnotám uzlů přiřadíme hodnoty $v_i + p_i$, kde $p_i = c(\deg u_i - 2)$, kde c je vhodná (ne příliš vysoká) kladná konstanta a deg u_i je stupeň daného uzlu nalezeného minimálního 1-stromu [6].

Může se stát, že při výpočtu dojde k zacyklení 2., 3. a 4. kroku. To je často způsobeno nedokonalou volbou čísla p_i. Proto je nutné třetí krok doplnit o mechanismus, který v takovém případě algoritmus zastaví. V případě, kdy ani nejdokonalejší volba penále, tj. čísel p_i, neumožní nalezení HK, se metoda penalizací kombinuje s metodou větví a mezí. Postup řešení se ovlivňuje tzv. vynucováním a zakazováním hran. Úloha se větví na dvě podúlohy, kdy zvolenou hranu buď zakážeme nebo si její použití naopak vynutíme a můžeme opět aplikovat metodu penalizací [6].

Tato metoda, stejně jako Kimova, vychází z nalezení minimální kostry (s následným rozšířením na minimální 1-strom). Vzhledem ke složitosti volby penále, resp. volby takového penále, při kterém by nedošlo k zacyklení algoritmu, je tato metoda komplikovanější pro zpracovaní. Její využití bych spatřoval spíše u ručního výpočtu menších sítí, kde máme možnost grafického zobrazení a tím i lepší přehlednosti výpočtu.

2.8 Metoda větví a mezí

Tato široce aplikovatelná a důležitá metoda opět nepatří mezi heuristické metody. Neprohledává množinu všech existujících přípustných řešení, ale pouze taková přípustná řešení, která prokazují největší zlepšení účelové funkce.

Metoda větví a mezí řeší úlohu dělením množiny přípustných řešení na menší podmnožiny a výpočtem horního, resp. dolního odhadu hodnot účelové funkce všech řešení v jednotlivých podmnožinách přípustných řešení. Odhady slouží k vyloučení těch podmnožin řešení, které optimální řešení neobsahují a slouží k určení nejnadějnější podmnožiny pro další větvení. Výpočet končí v okamžiku získání jednoprvkové podmnožiny, která obsahuje jediné řešení s hodnotou účelové funkce menší nebo rovné dolním odhadům hodnot účelové funkce na všech ostatních podmnožinách, které během výpočtu vznikly a nebyly dále děleny. Toto řešení je tedy optimálním, protože má nejmenší hodnotu účelové funkce ze všech přípustných řešení.

2.8.1 Principy a pravidla metody větví a mezí

Větvení množiny přípustných řešení na podmnožiny je založeno na dvou základních principech:

- ***princip větvení*** založen na postupném dělení množiny přípustných řešení na menší podmnožiny,
- ***princip odhadu mezí*** hodnot účelové funkce.

Podle těchto principů je i pojmenovaná tato metoda. Existuje několik pravidel, které jsou využívány při realizaci těchto principů:

1) Je-li při vylučování některé podmnožiny známo přípustné řešení úlohy, popř. hodnota jeho účelové funkce, je možné vyloučit z dalšího vyhledávání optimálního řešení všechny podmnožiny, které mají dolní odhad ú.f. větší než hodnota ú.f. dosud známého řešení.
2) Pro nalezení nejperspektivnější podmnožiny pro další větvení vycházíme z předpokladu, že největší naděje pro nalezení řešení s nejmenší hodnotou ú.f. je u podmnožiny s nejmenším dolním odhadem. Velikost dolního odhadu však nemusí být vždy úměrná minimální hodnotě ú.f. na řešeních dané podmnožiny.
3) Jestliže nastane případ, že dolní odhad nejperspektivnější podmnožiny se zvýší tak, že některá jiná podmnožina má dolní odhad nižší, pak je pro další větvení vybrána ta podmnožina, která má nejnižší odhad ze všech dosud nerozdělených podmnožin [6].

Proces větvení je možno znázornit pomocí kořenového stromu, kde kořenem bude množina *P* všech přípustných řešení, větvemi budou získané podmnožiny a listy, tj. koncové uzly nemající žádného následníka, budou tvořit dosud nerozdělené podmnožiny přípustných řešení.

Algoritmus založený na metodě větví a mezí je závislý na těchto faktorech:

1) postupu výpočtu dolního odhadu hodnot účelové funkce na množině přípustných řešení,
2) způsobu větvení určené podmnožiny,
3) na pravidlu zvolení podmnožiny pro větvení v dalším kroku.

Algoritmus využívá zpravidla jednoho ze dvou následujících způsobů vytváření a prohledávání stromu podmnožin přípustných řešení:

- prohledávání do hloubky s případným použitím zpětného návratu k nejbližšímu uzlu,
- usměrněného prohledávání [6].

Rozdíl v uvedených přístupech spočívá v pořadí, v jakém prohledáváme podmnožiny.

2.8.2 **Obecný algoritmus metody větví a mezí** (usměrněné prohledávání)

1) Inicializace: Definujeme hodnotu U dosud nejlepšího nalezeného řešení jako $U = +\infty$. Do seznamu listů přidáme množinu P všech přípustných řešení, vypočítáme dolní odhad f_D hodnot účelové funkce f na množině P.
2) Výběr nejperspektivnější podmnožiny: Je-li seznam listů prázdný, výpočet končí. Řešení odpovídající hodnotě U je optimální, v případě $U = +\infty$ řešení neexistuje. Je-li seznam listů neprázdný, pak vybereme ze seznamu množinu, která má nejmenší hodnotu odhadu f_D.
3) Větvení a stanovení dolní meze: Rozdělíme množinu listů na podmnožiny, vypočítáme jejich dolní meze a provedeme:
   a) Je-li nalezeno nejlepší řešení, aktualizujeme hodnotu U (dosud nejlepšího nalezeného řešení) hodnotou rovnou jeho účelové funkci.
   b) Jestliže krok a) nenastal, pak podmnožinu přidáme do seznamu listů a po zpracování všech podmnožin pokračujeme krokem 2) [6].

Z pohledu celočíselného programování představuje metoda větví a mezí základní přístup k řešení a patří mezi metody přesné. Míra přiblížení se k optimálnímu řešení je dána rozdílem mezi horní a dolní hodnotou účelové funkce. Máme dopravní síť, v němž každá hrana má ohodnocení $v_{ij} \geq 0$ nebo $v_{ij} = \infty$ (v případě, že hrana mezi uzly i a j neexistuje). Vznikne nám tzv. matice vzdáleností a její prvky vyjadřují vzdálenosti mezi jednotlivými uzly (místy). Matice vzdáleností nemusí být nutně symetrická.

Při požadavku na minimalizaci dopravní práce lze formulovat problém obchodního cestujícího jako úlohu celočíselného lineárního programování viz kapitola 2.1.3 [6].

Pro praktické výpočty se používá zejména ***Littleův algoritmus***, který je založený na principu prohledávání do hloubky s případným zpětným návratem k nejbližšímu uzlu. Množina řešení je definována množinou úseků, které budou zařazeny do minimální Hamiltonovy kružnice.

Metoda větví a mezí je metodou, která je často optimální a která se používá pro řešení problému i s větším počtem uzlů. Littleův algoritmus jako konkrétní přístup metody větví a mezí je vlastně algoritmem založeným na postupném větvení všech možných řešení. Narozdíl od rozhodovacího stromu však nevyhledává všechny možná řešení, ale pouze ty, která vykazují nejlepší zlepšení. LA je složitý především díky počtu operací, které je třeba s matici vzdáleností provést.

2.9 Zhodnocení a výběr metody pro řešení problému obchodního cestujícího

Po zhodnocení všech dílčích závěrů vyplývajících z uvedených metod jsem se rozhodl pro aplikaci metody rozhodovacího stromu, který bude pro větší počet uzlů zkombinován s metodou nejbližšího dosud nenavštíveného vrcholu. Pro použití RS rozhodl počet uzlů, pro který je aplikace navrhována. Navíc nespornou výhodou této metody je vyhledání optimálního řešení. V současných podmínkách zásobování firmy je tato metoda postačující. Dojde-li k nárůstu počtu míst potřebných zásobovat, stane se tato metoda pro časovou náročnost výpočtu nepoužitelnou. Proto je od vyšších počtů uzlů doplněna již zmíněnou metodou NDNV. Metoda NDNV je metodou vyhledávající suboptimální řešení, které je ale často lepší než původní.

3 Popis problému

3.1 Charakteristika firmy

Firma Kamahaj, která má sídlo nedaleko Opavy, se zabývá výrobou domácích cukrářských výrobků. Své produkty si firma rozváží sama, přičemž hlavními oblastmi zásobování jsou: Opavsko, Krnovsko a Vítkovsko. Vzhledem k částečným změnám, které se týkaly přesunu některých provozoven, do kterých směřovalo zásobování a které dále souvisely se vznikem nových komunikací či omezení provozu na stávajících komunikacích, vyvstala potřeba optimalizovat trasy, po kterých probíhá proces zásobování.

3.1.1 Způsob zásobování

Firma disponuje dvěma zásobovacími vozidly. První vozidlo je určené pro zásobování Opavy a blízkého okolí, druhé pak vyjíždí na delší vzdálenosti mimo region Opavska či opavský okres. Protože trasa, kterou projíždí druhé vozidlo, je už sama o sobě optimální (řidič volí nejkratší možné spojení míst především po hlavních komunikacích), bude problém spočívat v nalezení optimální trasy prvního vozidla. Trasa tohoto vozidla se skládá ze dvou okruhů. V prvním okruhu projíždí vozidlo jednotlivé místa (provozovny) pouze ve městě Opava. Druhý okruh je realizován po opětovném doplnění zboží. Vozidlo tedy vyjíždí opět ze stejného uzlu jako při prvním okruhu. Ve druhém okruhu probíhá zásobování provozoven lokalizovaných opět v Opavě s následným výjezdem do přilehlých obcí. Tento výjezd mimo město je realizován až po obsloužení všech provozoven ve městě. Většinu míst projíždí vozidla denně, některá místa projíždí jen v polovině pracovního týdne, o víkendu či zcela příležitostně. Tabulky 3.4 a 3.5 uvádějí četnost výjezdů k jednotlivým provozovnám za týden. Počet provozoven, které nejsou zásobovány v daný den, není více než dvě v každém z okruhů.

3.1.2 Dostupný software

Firmě této velikosti se nevyplatí investovat větší částky do softwaru vyvíjeným za komerčním účelem v oblasti optimalizačních problémů. S přihlédnutím k zaměření firmy je i toto značně neekonomické a vložené prostředky do koupi licence pro užívání mohou být investovány v jiných oblastech. Přehled programů pro řešení úloh operačního výzkumu je uveden v tab. 3.1. Je zřejmé, že uvedené programy neřeší jenom problém zásobování

(POC), ale jejich použití najdeme v celé řadě optimalizačních úloh. Společným znakem uvedených programů je jejich komerční využití. Jsou poskytovány za stanovenou cenu, která závisí na složitosti a verzi systému a na možnosti zpracování obtížnějších problémů. Je třeba podotknout, že firma by koupí softwaru získala možnost zpracování i jiných problémů, ale nabízí se otázka, zda by tuto možnost využila.

Název produktu	Stručná charakteristika	Výhody / Nevýhody
STORM – systém pro řešení úloh operačního výzkumu	Programový prostředek pro řešení a analýzu v praxi nejčastěji používaných úloh z oblasti operačního výzkumu a statistiky. Obsahuje celkem 16 modulů, které umožňují řešit úlohy lineárního programování, řízení projektů, teorie zásob, teorie hromadné obsluhy atd.	Uživatelsky velmi přívětivý, řeší široké spektrum úloh. Nevýhody: cena, angl. jazyk.
Optimalizační systém **MOR/LP**	Určen pro řešení optimalizačních úloh, z hlediska rozsahu řešených úloh a numerické přesnosti výpočtů je určen spíše pro výpočetní ilustraci algoritmů.	Nenáročnost na technické parametry. Nevýhody: jazyk, cena, neumožňuje řešení větších úloh.
Optimalizační systém **LINDO**	Jeden z nejpoužívanějších a výpočetně nejspolehlivějších komerčně šířených optimalizačních systémů. Dodává se v různých verzích, které jsou vhodné pro řešení úloh různých náročností na výpočet. Od tohoto hlediska se odvíjí také cena.	Snadné ovládání. Nevýhodou je vysoká cena a angl. jazyk.
Profesionální optimalizační systém **XA**	Profesionální produkt pro řešení úloh lineárního programování se spojitými a diskrétními proměnnými. V současné době existuje málo komerčně šířených optimalizačních systémů, které by byly svými možnostmi, rychlostí a přesností výpočtů s XA srovnatelné.	Nejméně uživatelsky přátelský, vyžaduje větší uživatelské znalosti a zkušenosti při práci s výpočetní technikou. Další nevýhodou je nejvyšší cena a jazyk.

Tab. 3.1: Přehled dostupného softwaru pro optimalizační úlohy Zdroj: Vlastní zpracování [8]

3.2 Požadavky na aplikaci

Firma požaduje, aby stávající trasa, kterou při svém zásobování projíždí první vozidlo, byla zoptimalizována a to z hlediska vzdálenostního. Jedná se o optimalizaci dvou okruhů v Opavě, které budou dále podrobněji popsány. Se vzdálenostní optimalizací souvisí nákladová a časová úspora.

Navržená aplikace by měla být plně kompatibilní s operačním systémem Windows. Měla by splňovat požadavek snadné uživatelské ovladatelnosti a být navržena tak, aby při případných změnách byla schopna i nadále, po změně vstupních údajů, plnohodnotně plnit svoji funkci. Je třeba také přihlédnout k možnosti změny počtu míst (uzlů), do kterých směřuje zásobování. Po konzultaci s firmou bylo stanoveno, že počet uzlů by neměl překročit více jak 30.

Vzhledem k požadavkům, které firma od navržené aplikace očekává, bylo zvoleno řešení v programu Excel jako součást balíku MS Office, ke kterému má firma zakoupenou licenci a může ho tedy bez problému využívat. Excel nabízí příjemné uživatelské prostředí a je vhodným nástrojem potřebným k vytvoření aplikace. Při tvorbě aplikace se využívají jak neprogramové nástroje tak i jazyk Visual Basic for Applications (VBA). Prostředí Excelu respektuje všechny požadavky kladené na aplikaci, nabízí snadnou manipulaci s daty a je v tomto ohledu dobrou volbou.

3.3 Charakteristika současného stavu

Tab. 3.2 a 3.3 uvádějí seznamy jednotlivých provozoven, které je třeba obsloužit. Firma nezamýšlí větší rozšiřování svých současných výrobních kapacit, ať už jde o objemové rozšiřování nebo také o teritoriální zvětšovaní zásobovací oblasti. Navíc je zřejmé, že u výroby tohoto typu se zásobování na delší vzdálenosti finančně nevyplácí, protože substituentem zásobování jsou provozovny lokalizované v daných oblastech.

Tab. 3.2 zachycuje seznam uzlů, které je třeba projet v prvním okruhu. První okruh obsahuje celkem 13 uzlů, ale vzhledem ke konkrétním podmínkám může dojít k částečné redukci tohoto počtu. Uzel 6 a 7 představuje dvě různé provozovny, avšak vzdálenost těchto provozoven je jen několik málo metrů a vozidlo se při obsloužení 7. uzlu musí vrátit přes 6. uzel. V tomto případě budeme uzel 6 a 7 považovat za jediný a v dalším výkladu jej budeme označovat jako 6. Stejný případ nastává také u uzlů 12 a 13, které budou představovat samotný uzel 11 (došlo ke snížení počtu o jeden důsledkem redukce sedmého uzlu).

Adresa provozoven	Označení	Upravené označení	Četnost výjezdů
Podolská 321, Hradec nad Moravicí	1	1	Výchozí uzel
Rybova 26, Opava	2	2	Každý den
Ostrožná 30, Opava	3	3	Každý den
Hrnčířská 14, Opava	4	4	Každý den
Horní náměstí 29, Opava	5	5	Každý den
Polní 12, Opava	6	6	Každý den
Polní 13, Opava	7	-	Vybrané dny
U lučního mlýna 29, Opava 5	8	7	Dvakrát týdně
Hlučínská 49, Opava 5	9	8	Každý den
Ratibořská 119, Opava 5	10	9	Každý den
Vrchní 37, Opava 5	11	10	Každý den
Krnovská 165, Opava 7	12	11	Každý den
Krnovská 153a, Opava 7	13	-	Jedenkrát týdně

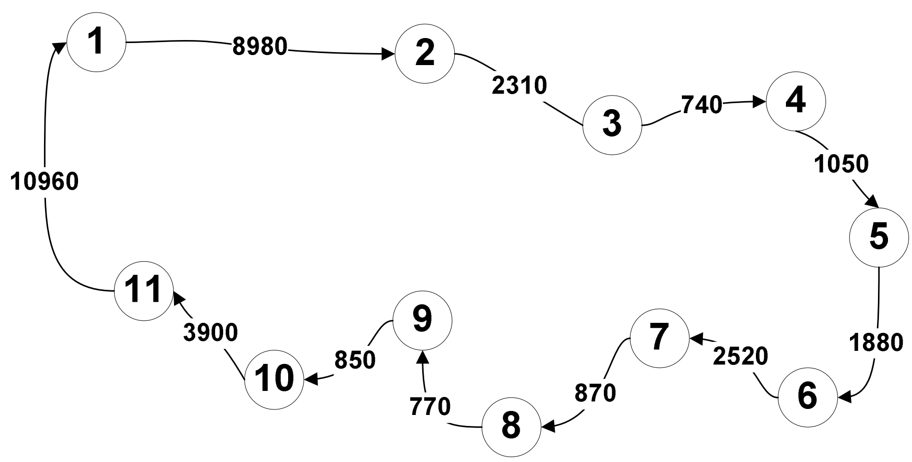
Tab. 3.2: Seznam provozoven 1. okruhu

Adresa provozoven	Označení	Četnost výjezdů
Podolská 321, Hradec nad Moravicí	1	Výchozí uzel
Olomoucká 88, Opava	2	Každý den
Olomoucká 29, Opava	3	Každý den
Nám. Republiky 3, Opava	4	Každý den
Rybářská 27, Opava	5	Každý den
Nám. Osvoboditelů 3, Opava	6	Každý den
Hozovo nábřeží, Opava 5	7	Každý den
Ratibořská 99a, Opava 5	8	Dvakrát týdně
Osvobození 47, Opava 6	9	Jedenkrát týdně
Tyršova 20, Opava	10	Příležitostně

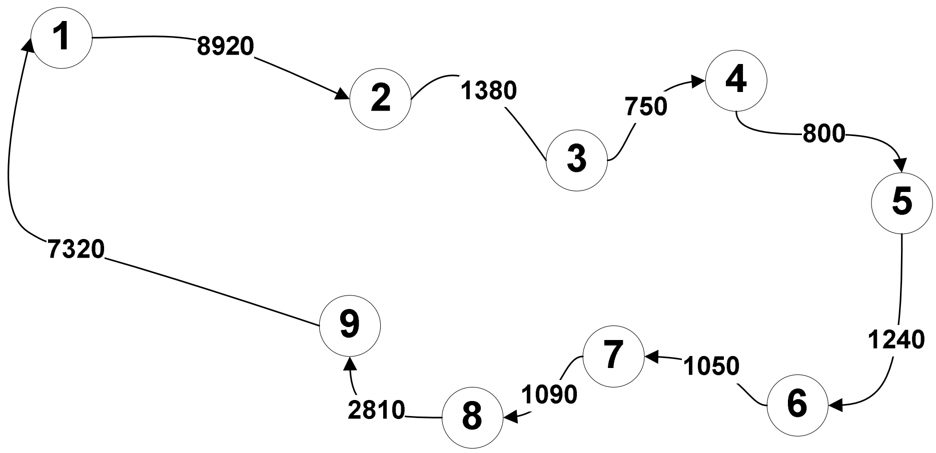
Tab. 3.3: Seznam provozoven 2. okruhu

Tab. 3.3 uvádí seznam míst, které projede vozidlo ve druhém okruhu. Desátý uzel je zde uveden jen pro úplnost, firma zde provádí zásobování zcela výjimečně na základě domluvy. Proto bude tento okruh v dalším řešení úlohy vypočítáván bez projetí tímto uzlem. Jednotlivé vzdálenosti z tohoto i do tohoto 10. uzlu jsou v tab. 3.5 uvedeny. Firma si ale sama zvolí, zda tento uzel zahrne do výpočtu nebo ne a to takovým způsobem, že si zvolí rozměr matice 10 (v případě, že uzel nechce projet) nebo 11 v opačném případě. To je umožněno díky poslední pozici uzlu v matici. V případě, kdy je rozměr matice 10, je jedenáctý řádek a sloupec při výpočtu vynechán.

Obr. 3.1 a 3.2 zachycují síť grafu, který znázorňuje dvě uzavřené trasy, které projíždí vozidlo při zásobování. Počáteční uzel, tedy místo odkud vozidlo vyjíždí na svou trasu, je označen číslem jedna. Při prvním okruhu vozidlo navštíví 10 míst a vrací se zpět do firmy, odkud po doplnění zboží opětovně vyráží na další okruh. Při tomto druhém okruhu vozidlo opouští po obsloužení všech provozoven město a jede po nejbližší hlavní komunikaci navštívit další místa, která jsou lokalizovaná mimo město. Tento výjezd je uskutečněn mezi 9. a 1. uzlem. Desátý uzel vozidlo projíždí jen příležitostně, proto bude problém řešen bez obsloužení tohoto uzlu, i když je uveden v tabulce uvádějící seznam provozoven. Provozovny, které vozidlo obsluhuje mimo město, nejsou na obr. 3.2 znázorněny, neboť vozidlo při svém výjezdu opouští město na dané komunikaci, na kterou se při svém návratu vrací. Tento okruh (smyčka) je již minimalizován z hlediska vzdálenostního, a proto bylo od něho ve znázorněné síti i v dalším výkladu abstrahováno.



Obr. 3.1: Původní posloupnost uzlů 1. okruhu



Obr. 3.2: Původní posloupnost uzlů 2. okruhu

Grafy na obr. 3.1 a 3.2 znázorňují dvě hamiltonovské kružnice - je splněna podmínka navštívení každého uzlu právě jednou a současně existuje mezi výchozím a konečným uzlem platná cesta. Uvedená síť byla doplněna na úplnou síť, tzn. že každý uzel je spojen cestou se všemi ostatními uzly. Dostali jsme úplný graf, který neobsahuje žádné smyčky (nepotřebujeme cestu, která vede z jednoho uzlu do téhož). Jestliže si označíme počet uzlů jako n, pak takto upravený graf bude obsahovat právě $(n^2 - n)$ různých cest, pozn. cesta z uzlu 1 do uzlu 2 nemusí být stejně dlouhá jako cesta z uzlu 2 do uzlu 1. Jedná se o pravidelný graf, který má všechny uzly stejného stupně $(n - 1)$. Jednotlivé vzdálenosti mezi danými uzly prvního, resp. druhého okruhu uvádí tab. 4.2, resp. 4.3. Vzdálenosti jsou uvedeny v metrech a jsou zaokrouhleny na desítky metrů.

Takto upravený graf, doplněný o všechny cesty z daného uzlu do všech ostatních v okruhu, splňuje nutné i postačující podmínky existence Hamiltonovy kružnice. Je třeba uvést, že při hledání cest spojující všechny ostatní uzly grafu, byly brány do úvahy tyto hlediska:

1) minimalizace vzdálenosti mezi dvěma sousedními uzly,
2) bod 1) byl upraven o minimalizaci časovou při současné neměnnosti či minimálnímu nárůstu vzdálenostní minimalizace,
3) byly respektovány všechny současné omezení vyplývající z místní úpravy dopravy.

	kam										
x / y	1	2	3	4	5	6	7	8	9	10	11
1	0	8980	8560	8370	8980	9190	9760	10020	9250	9570	10960
2	8980	0	2310	2900	2720	3460	4030	4370	3610	3660	3620
3	8560	2090	0	740	1240	1800	2100	2350	1580	1900	3350
4	8370	2440	260	0	1050	1610	1930	2200	1440	1740	3800
5	9120	2650	1240	550	0	1880	2330	2610	1840	1850	3610
6	9190	3540	2190	2030	2230	0	2520	2760	1990	2310	5140
7	9760	4120	2090	1930	2330	2520	0	870	530	1050	4970
8	10020	4370	2360	2200	2580	2770	870	0	770	1300	5190
9	9250	3590	1590	1440	1810	2000	530	770	0	850	4560
10	9570	3640	1990	1730	1700	2320	1050	1280	640	0	3900
11	10960	2930	3340	3830	3610	5140	4970	5190	4560	3900	0

odkud

Tab. 3.4: Vzdálenosti prvního okruhu

	kam									
x / y	1	2	3	4	5	6	7	8	9	10
1	0	8920	8010	8370	8980	8460	9400	9060	7320	7550
2	8920	0	1380	2130	2380	3200	3280	3580	3930	2640
3	8010	1380	0	750	1000	1880	1810	2380	2790	1370
4	8370	2130	750	0	800	320	1360	1940	2730	1430
5	8980	2380	1000	800	0	1240	1160	1880	3380	2100
6	8460	3200	1820	320	1240	0	1050	740	2210	1420
7	9540	3260	1750	1360	1160	1220	0	1090	3150	2410
8	9060	3400	2130	2220	1880	740	1090	0	2810	2250
9	7320	3930	2790	2730	3380	2210	3110	2810	0	1790
10	7850	2210	870	970	1620	830	1740	1440	1930	0

odkud

Tab. 3.5: Vzdálenosti druhého okruhu

4 Aplikace vybrané metody

Při volbě algoritmu, který byl vybrán pro řešení daného problému, jsem vycházel z předpokladu, že zásobování probíhá ve dvou po sobě následujících okruzích, které v obou případech začínají a končí ve stejném uzlu. Protože je problém rozdělen na dva okruhy, poklesl počet uzlů v jednotlivých dílčích okruzích s porovnáním s možností zásobování při jediném výjezdu.

Tento způsob zásobování firma zamýšlí ponechat i nadále. Důvodem je smluvený čas příjezdu vozidla do prodejny a podle toho, v jakém časovém intervalu je požadované zboží do prodejny přivést, je daný uzel zařazen do prvního nebo druhého okruhu.

4.1 Aplikace metody rozhodovacího stromu

Počet uzlů v jednotlivých okruzích je natolik malý, že byla zvolena metoda, která spočívá v prohledání všech možných uzavřených cest mezi danými uzly, metoda rozhodovacího stromu. Při řešení tak dospějeme k optimálnímu řešení narozdíl od heuristických metod.

Je zřejmé, že u metody RS roste náročnost výpočtu s každým dalším uzlem a že doba potřebná pro výpočet neúměrně roste. Je-li počet uzlů roven n, pak existuje $(n-1)!$ různých větví, které vedou ze zvoleného počátečního uzlu do téhož koncového uzlu, přičemž každý uzel bude zastoupen právě jednou. Přestože je počet větví RS roven číslu $(n-1)!$, bude počet prohledávaných kombinací menší. Při prvním průběhu algoritmu založeném na postupném prohledávání jednotlivých kombinací je nalezena určitá hodnota. Tato hodnota většinou není hodnotou optimální a odpovídá posloupnosti uzlů $x_{1,2}, x_{2,3}, x_{3,4}, \ldots, x_{n-1,n}$. Následně je tato první nalezená hodnota uložena do proměnné, která je pak v dalším prohledávání srovnávána s aktuální hodnotou právě prohledávanou. Jestliže je aktuálně prohledávaná hodnota větší jak hodnota proměnné, algoritmus danou větev dále nepropočítává a pokračuje v dalším prohledávání jiné kombinace uzlů. Jestliže je naopak nalezena hodnota menší než hodnota uložená v proměnné, stane se novou hodnotou optimální a testování podmínky je prováděno s touto proměnnou.

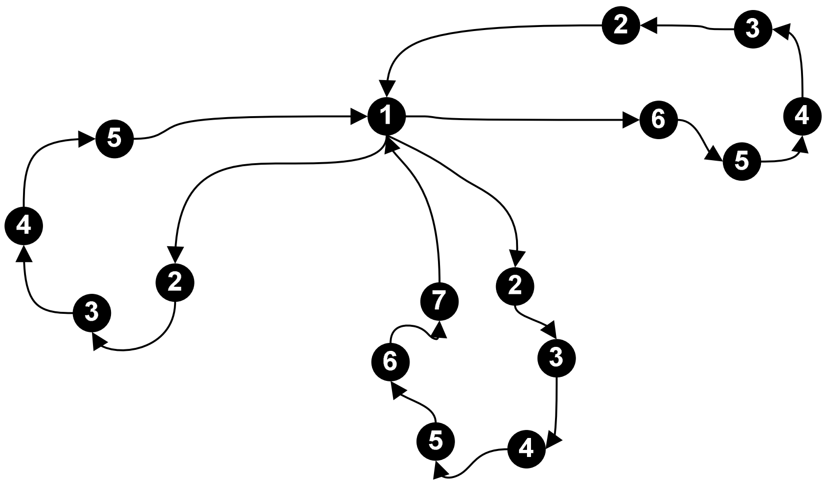
Jestliže tuto metodu rozhodovacího stromu aplikuji na uvedený problém, nic nebrání jejímu použití, protože operujeme s malým počtem uzlů. Tuto metodu je možné realizovat do počtu třinácti, maximálně čtrnácti uzlů (závisí na tom, jak výkonným hardwarem disponujeme). Jestliže si spočítáme počet kombinací, které nastanou při čtrnácti uzlech je zřejmé, že větší počet uzlů není obyčejný stolní počítač schopný zvládnout. V současném způsobu zásobování (počtu uzlů) je schopný tento algoritmus běžet, aniž by došlo k neúnosně dlouhé době výpočtu.

Může však nastat situace, kdy firma rozšíří počet stávajících uzlů, dojde ke změně polohy uvedených uzlů nebo se firma rozhodne pro jiný způsob zásobování. Proto je metoda rozhodovacího stromu od vyšších počtů uzlů nahrazena algoritmem pro nalezení minimální hamiltonovské kružnice metodou nejbližšího dosud nenavštíveného vrcholu. Tato metoda je schopná běžet i při vyšších počtech uzlů s dostatečnou rychlostí zpracování. Nevýhodou oproti rozhodovacímu stromu je, že poskytuje pouze suboptimální řešení. Často je však toto řešení lepší než původní neoptimalizované.

4.1.2 Návrh k použití metody rozhodovacího stromu pro větší počet uzlů

Metoda rozhodovacího stromu představuje přesné a tedy optimální řešení, a je proto výhodné tuto metodu uplatnit i při větším počtu uzlů než je třináct, popř. čtrnáct. Jak už bylo uvedeno, algoritmus rozhodovacího stromu je velmi náročný na výpočet při větších počtech uzlů. Tomu se dá však předejít způsobem, který spočívá v rozdělení problému do dílčích podproblémů a na tyto dílčí podproblémy následně aplikovat zmíněný algoritmus.

K vytvoření podproblému dojde rozdělením celkového počtu uzlů na menší počet tak, abychom zamezili nemožnosti výpočtu s velikým množstvím uzlů. Jestliže existuje větší koncentrace uzlů v jednom místě, přičemž po projetí všech uzlů v tomto místě se nevracíme do výchozího bodu, ale opět vyrážíme na další dílčí okruh opět s větším výskytem uzlů, pak tyto menší okruhy můžeme řešit zvlášť. Je potřeba najít pro tyto okruhy takový uzel, který splňuje podmínky nejkratší vzdálenosti z daných okruhů. Tento uzel pak bude výchozím uzlem daných okruhů s redukovaným počtem uzlů. Obr. 4.1 znázorňuje rozdělení velkého okruhu na tři menší okruhy. Na těchto třech dílčích okruzích hledáme minimální trasu metodou rozhodovacího stromu.



Obr. 4.1: Rozdělení problému na dílčí podproblémy Zdroj: Vlastní zpracování

4.2 Popis vytvořené aplikace

Jednotlivé matice – tedy matice prvního a druhého okruhu – jsou zaznamenány v sešitu s názvem obch_cestujici.xls. Po spuštění aplikace se automaticky objeví nabídka, která uživatele navede k provedení jednoho ze tří kroků.

První možností pro uživatele je spuštění nápovědy k aplikaci. Pro nápovědu je třeba kliknout na tlačítko Help. Objeví se dialogové okno, které uživatele informuje o možnostech tohoto programu a slouží jako návod k obsluze aplikace .

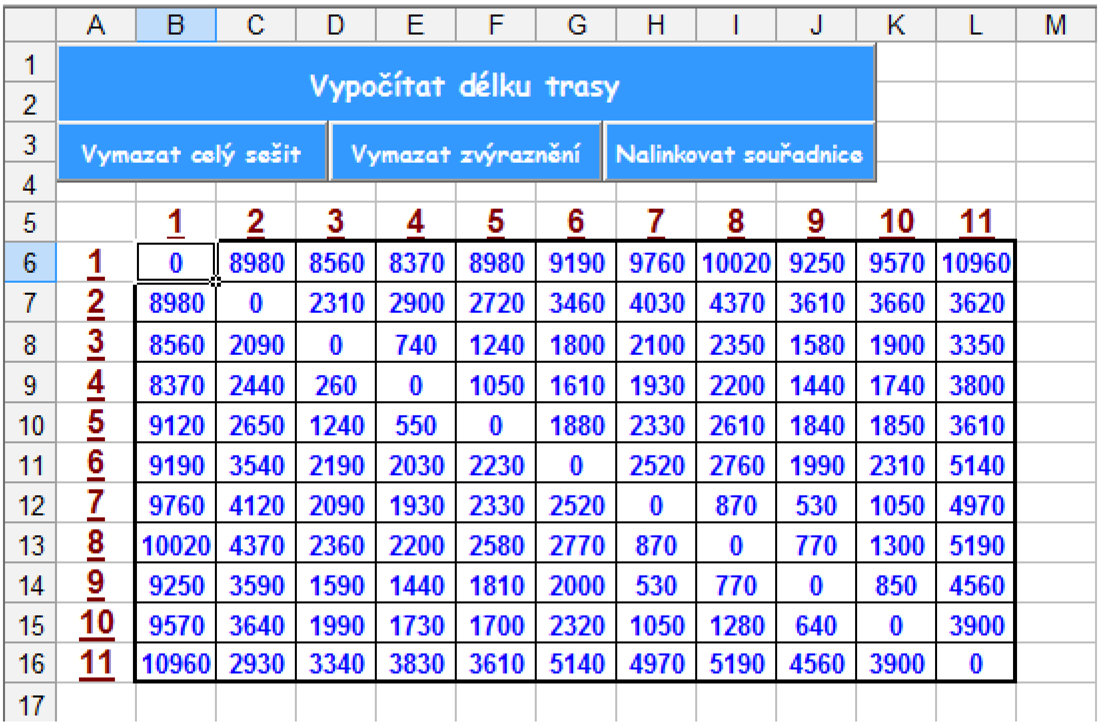
Další z možností je tlačítko Storno, které po kliknutí uzavře aplikaci. Jestliže dojde ke změně údajů, otevře se nabídka pro uložení.

Poslední variantou je spuštění samotné aplikace.

Protože zápis více okruhů v jediném listu by byl značně nepřehledný a komplikovaný, jsou jednotlivé matice zaznamenány na samostatných listech, které disponují svojí vlastní nabídkou. Listy jsou uspořádány a pojmenovány podle okruhů. Aplikace obsahuje celkem čtyři listy. V prvním listu „první okruh" je matice prvního okruhu, druhý list s názvem „druhý okruh" obsahuje matici druhého okruhu, třetí list „jiný okruh" slouží pro výpočet dosud firmou nedefinovaného (dosud neexistujícího) okruhu.
Zde je tedy nabídnut firmě prostor pro výpočet dalšího okruhu, aniž by došlo ke smazání údajů v prvních dvou listech. Poslední čtvrtý list pojmenovaný „řešení" slouží pro výpisy

délky trasy a posloupnosti míst, které je třeba zásobovat. Tento způsob výpisu umožňuje uživateli vytisknout požadované řešení bez dalších nepotřebných údajů (vzdálenostní matice a tabulky udávající adresy provozoven).

Ukázka zápisu matice je zachycena na obr. 4.2.



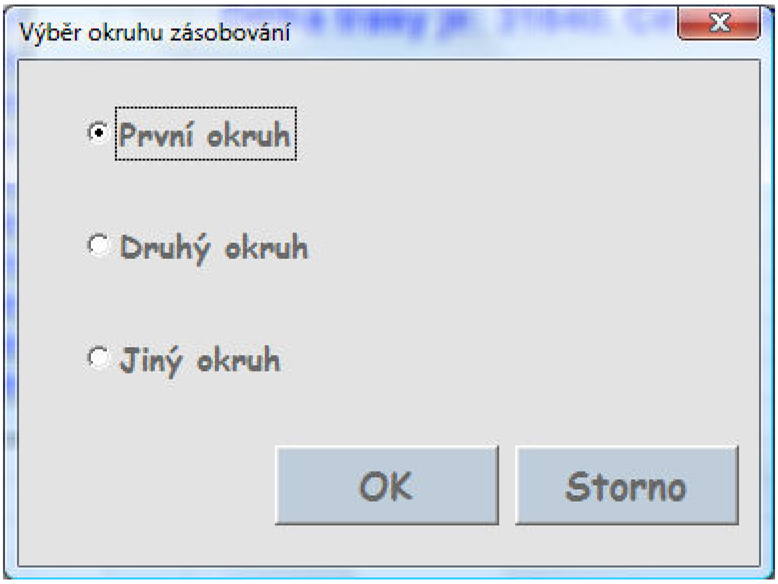
Vypočítat délku trasy

Vymazat celý sešit | Vymazat zvýraznění | Nalinkovat souřadnice

	1	2	3	4	5	6	7	8	9	10	11
1	0	8980	8560	8370	8980	9190	9760	10020	9250	9570	10960
2	8980	0	2310	2900	2720	3460	4030	4370	3610	3660	3620
3	8560	2090	0	740	1240	1800	2100	2350	1580	1900	3350
4	8370	2440	260	0	1050	1610	1930	2200	1440	1740	3800
5	9120	2650	1240	550	0	1880	2330	2610	1840	1850	3610
6	9190	3540	2190	2030	2230	0	2520	2760	1990	2310	5140
7	9760	4120	2090	1930	2330	2520	0	870	530	1050	4970
8	10020	4370	2360	2200	2580	2770	870	0	770	1300	5190
9	9250	3590	1590	1440	1810	2000	530	770	0	850	4560
10	9570	3640	1990	1730	1700	2320	1050	1280	640	0	3900
11	10960	2930	3340	3830	3610	5140	4970	5190	4560	3900	0

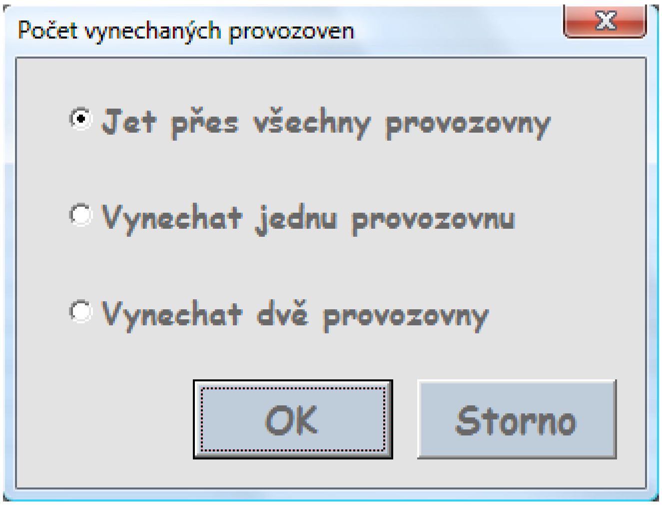
Obr. 4.2: Zápis matice vzdáleností

V dalším kroku si uživatel může vybrat okruh, který chce optimalizovat. Na výběr má tři možnosti: první okruh, druhý (menší) okruh či zcela jiný okruh (obr. 4.3). Poslední možnost je zde uvedena pro případ nárůstu počtu okruhů, spojení okruhů do jednoho velkého (tato možnost je zatím uživatelem zamítnuta) nebo přesunu určité provozovny do jiného okruhu vlivem dočasné změny termínu zásobování. Přitom budou zachovány okruhy v prvním a druhém listě.



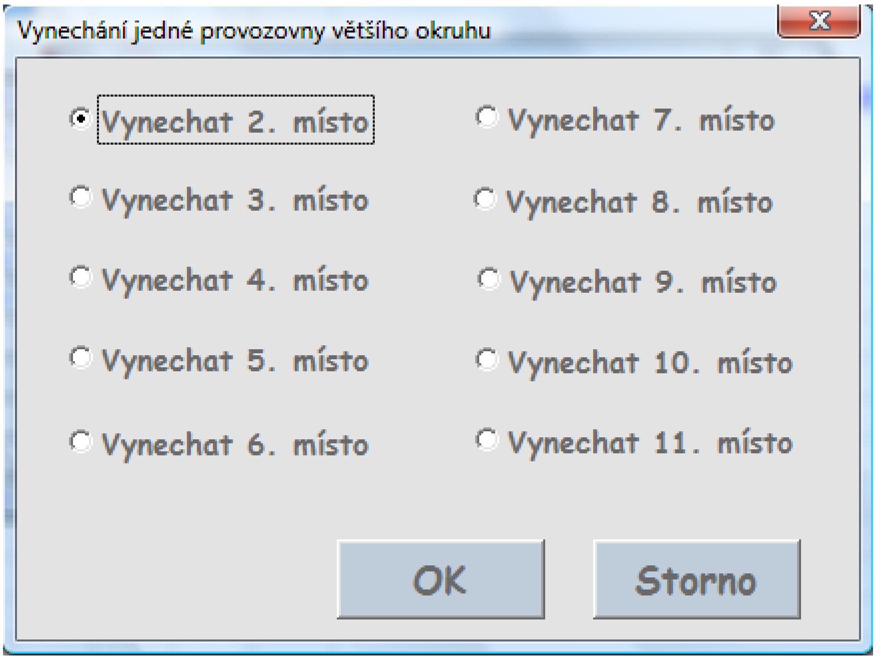
Obr. 4.3: Výběr okruhu zásobování

Je-li vybrán větší či menší okruh, pak je možné tyto okruhy vypočítat pro všechny zaznamenaná místa v prvním a druhém listu (obr. 4.4).



Obr. 4.4: Vynechání počtu provozoven

Protože se však může stát, že některá z provozoven bude daný den z nějakého důvodu uzavřena a tím dojde ke změně některého z okruhů, je uživateli nabídnuta možnost vynechání některého z jedenácti, v případě druhého okruhu desíti uzlů (obr. 4.5). Dojde tedy k redukci základního počtu obsluhovaných míst a výpočtu nové délky a kombinace uzlů původní trasy.



Obr. 4.5: Vynechání konkrétní provozovny

Uživatel má možnost vynechání jednoho nebo současně dvou různých uzlů, jak ukazuje obr. 4.6 v příloze č. 2. Větší počet neprojetých uzlů v daný den se nepředpokládá. Jestliže dojde ke zrušení provozovny nebo jejímu přesunu do jiného místa, musí se tato změna upravit v matici vzdáleností přímo v daném listu. Velikost matice je zadávána formou dialogového okna, do kterého je zapsáno počet míst k obsloužení.

Aplikace je ošetřena proti zadávání nepřípustných údajů, jako jsou nepovolené velikosti matic nebo zadání nepřípustného znaku, popř. jsou-li velikosti zadané matice větší než velikosti skutečně zapsané. Na obr. 4.6 (příloha č. 2) je znázorněn nepovolený výběr dvou stejných uzlů a s následným zobrazením kontrolní hlášky.

5 Hodnocení přínosu řešení

5.1 Délka původní trasy

Obr. 3.1, resp. 3.2 zachycuje původní trasu zásobování prvního, resp. druhého okruhu. Trasa šla přes jednotlivé uzly v posloupnosti 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, resp. 1, 2, 3, 4, 5, 6, 7, 8, 9. Délka okruhů byla před optimalizací dána součtem uzlů v uvedené posloupnosti. Trasy byly projížděny v logické návaznosti jednotlivých míst, nikdy však nebyla provedena optimalizace trasy zásobování.

U prvního okruhu byla uvedená trasa dlouhá 34 830 metrů, druhý okruh pak měřil 25 360 metrů.

5.2 Délka trasy po optimalizaci, interpretace řešení

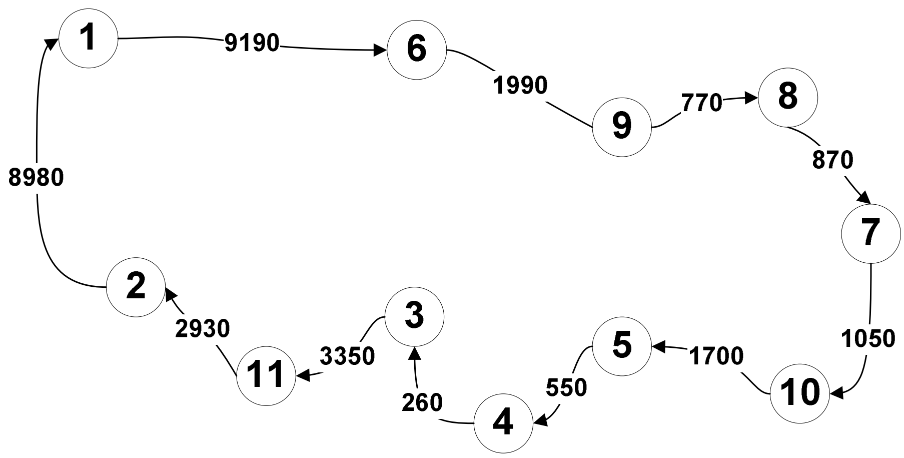
Po optimalizaci došlo ke zkrácení trasy prvního okruhu o 3 190 metrů. První trasa je tedy po optimalizaci dlouhá 31 640 metrů (obr. 5.1). Procentní zlepšení je 9,2 %, přičemž velkou roli, která zabraňuje vyššímu číslu procentního zlepšení, hraje první a poslední úsek takto nalezené minimální trasy. Tyto dva úseky představují největší podíl na celkové vzdálenosti a dochází tím ke zkreslení procentního vyjádření zlepšení. Obr. 5.2 znázorňuje itinerář cesty s konkrétními adresami provozoven, který si může řidič před výjezdem na cestu vytisknout. Nový okruh zásobování včetně vzdáleností mezi jednotlivými uzly je graficky znázorněn na obr. 5.3.

Délka trasy je 31640 metrů. Cesta vede: 1, 6, 9, 8, 7, 10, 5, 4, 3, 11, 2, 1.

Obr. 5.1: Výpis délky trasy pro první okruh

	A	B
1		
2	1	Podolská 321, Hradec nad Moravicí
3	2	Polní 12, Opava
4	3	Ratibořská 119, Opava 5
5	4	Hlučínská 49, Opava 5
6	5	U lučního mlýna 29, Opava 5
7	6	Vrchní 37, Opava 5
8	7	Horní náměstí 29, Opava
9	8	Hrnčířská 14, Opava
10	9	Ostrožná 30, Opava
11	10	Krnovská 165, Opava 7
12	11	Rybova 26, Opava
13	12	Podolská 321, Hradec nad Moravicí
14		

Obr. 5.2: Itinerář cesty prvního okruhu



Obr. 5.3: Grafické znázornění posloupnosti uzlů 1. okruhu po optimalizaci

Druhý okruh je po optimalizaci dlouhý 24 370 metrů, jak je vidět na obr. 5.4 (výstup aplikace). Zkrácení trasy oproti původní je o 990 metrů, což představuje 3,9 procentní zlepšení. Uvědomíme-li si ale fakt, že největší nárůst délky trasy představuje úsek mezi výchozím (prvním) uzlem a prvním obslouženým uzlem, resp. úsek mezi posledním obslouženým uzlem a konečným (prvním) uzlem je toto zlepšení velké. Uvedené úseky představují okolo 2 / 3 délky celkové trasy.

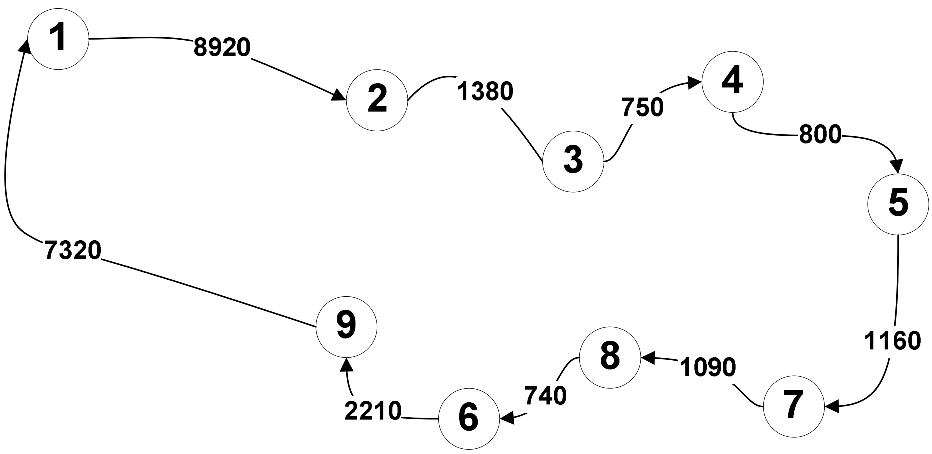
Obr. 5.4 ukazuje, že existují dvě různé cesty, které mají stejnou délku. Firma si tak může vybrat, která trasa je pro ni vhodnější, resp. kterou posloupnost uzlů bude při zásobování preferovat. Uvedené trasy se liší pouze směrem projíždění, to je způsobeno stejnou vzdáleností mezi uspořádanou dvojicí uzlů dané posloupnosti (např. uspořádané dvojice $[u_{12}, u_{23}]$ a $[u_{21}, u_{32}]$ mají stejnou délku).

Délka trasy je 24370 metrů. Cesta vede: 1, 2, 3, 4, 5, 7, 8, 6, 9, 1.
Délka trasy je 24370 metrů. Cesta vede: 1, 9, 6, 8, 7, 5, 4, 3, 2, 1.

Obr. 5.4: Výpis délky tras pro druhý okruh

	A	B	C	D	E	F
1						
2	1	Podolská 321, Hradec nad Moravicí				Podolská 321, Hradec nad Moravicí
3	2	Olomoucká 88, Opava				Osvobození 47, Opava 6
4	3	Olomoucká 29, Opava				Nám. Osvoboditelů 3, Opava
5	4	Nám. Republiky 3, Opava				Ratibořská 99a, Opava 5
6	5	Rybářská 27, Opava				Hozovo nábřeží, Opava 5
7	6	Hozovo nábřeží, Opava 5				Rybářská 27, Opava
8	7	Ratibořská 99a, Opava 5				Nám. Republiky 3, Opava
9	8	Nám. Osvoboditelů 3, Opava				Olomoucká 29, Opava
10	9	Osvobození 47, Opava 6				Olomoucká 88, Opava
11	10	Podolská 321, Hradec nad Moravicí				Podolská 321, Hradec nad Moravicí
12						

Obr. 5.5: Itinerář cesty druhého okruhu



Obr. 5.6: Grafické znázornění posloupnosti uzlů 1. okruhu po optimalizaci

Závěr

Cílem bakalářské práce bylo seznámení s možnými přístupy k řešení problému obchodního cestujícího, provedení jejich vzájemné komparace a vytvoření aplikace umožňující řešení POC.

Práce je postavena na definování jednotlivých přístupů k POC s jejich následným zhodnocením a výběrem nejvhodnějšího z nich. Uvedený výčet přístupů, kterým se v práci věnuji, není vyčerpávající. Zahrnuje nejběžnější a nejpoužívanější metody, kterými jsou: metoda nejbližšího dosud nenavštíveného vrcholu, Kimova metoda, metoda rozhodovacího stromu, metoda penalizací a metoda větví a mezí.

Metoda NDNV je metodou, která neposkytuje optimální řešení (pouze výjimečně - při malém počtu uzlů - může být řešení optimální). Vyhledané řešení je suboptimální, ale často lepší než původní. Metoda je jednoduchá na aplikaci, jak pro ruční výpočet, tak pro počítačové zpracování.

Metoda RS, která spočívá v postupném prohledávání všech možných kombinací cest a vyhledání nejkratší možné kombinace, je metodou dávající optimální řešení. Podstatnou nevýhodou většího uplatnění RS je časová náročnost výpočtu. Nicméně pro menší počty uzlů je ideálním řešením. Vzhledem k počtu možných kombinací je tato metoda vhodná především pro počítačové zpracování, protože s každým dalším uzlem roste počet možných variant exponenciálním způsobem.

Další dvě metody - Kimova a metoda penalizací - využívají poznatků teorie grafů. Obě metody jsou heuristické, nalezené řešení je suboptimální. KM vyžaduje složité vyhledávání uzlů zastoupených v minimální kostře více než jednou, u metody penalizací je zase obtížný způsob volby vhodného penále, při kterém by nedošlo k zacyklení. Využití těchto dvou metod bych spatřoval u menších sítí s možností grafického zobrazení.

Poslední metodou, které se v práci věnuji je metoda větví a mezí. V praxi se používá především Littleův algoritmus, který je založený na principu prohledávání do hloubky s případným zpětným návratem k nejbližšímu uzlu. Tento algoritmus je náročný na počet operací, které je třeba s maticí vzdáleností provést.

Po konečném zhodnocení byly v aplikaci vybrány následující metody:

- metoda rozhodovacího stromu,
- metoda nejbližšího dosud nenavštíveného vrcholu.

Aplikace byla vytvořena v Excelu s použitím jazyka VBA a dalších neprogramovatelných prvků. Při tvorbě byl kladen důraz na snadnou uživatelskou obsluhu, na zřetel byla brána možnost změny počtu a rozmístění (tedy vzájemné vzdálenosti) jednotlivých míst. Program umožňuje také vynechání některých uzlů s vypočtením nového okruhu.

V prvním okruhu, které vozidlo urazí při své trase a přitom nevynechá jedinou provozovnu, došlo ke snížení celkové vzdálenosti o 3 190 metrů, což představuje 9,2 procentní zlepšení. V druhém okruhu dojde ke zkrácení trasy o 990 metrů, tedy o 3,9 %. V průměru dojde tedy ke ztrácení obou tras o 6,6 %.

Řidiči vozidla se zkrátí především doba potřebná k obsloužení míst, dojde také k částečné úspoře nákladů vlivem menší spotřebě pohonných hmot.

Seznam použité literatury

knižní zdroje

[1] JABLONSKÝ, J. *Operační výzkum. Kvantitativní modely pro ekonomické rozhodování*. 1. vyd. PRAHA: Professional Publishing, 2002. 323 s. ISBN 80-86419-23-1

[2] KUČERA, L. *Kombinatorické algoritmy*. 2. nezm. vyd. PRAHA: Státní nakladatelství technické literatury, 1989. 286 s.

[3] LAGOVÁ, M., JABLONSKÝ, J. *Lineární modely*. 1. vyd. PRAHA: Vysoká škola ekonomická v Praze, 1999. 284 s. ISBN 80-7079-959-5

[4] LAUBER, J., JABLONSKÝ, J. *Programy pro matematické modelování I.* 1. přeprac. vyd. PRAHA: Vysoká škola ekonomická v Praze, 1997. 233 s. ISBN 80-7079-296-5

[5] ŘÍMÁNEK, J., ZONKOVÁ, Z., POŠTOVÁ, E., MORAVCOVÁ, E., HANČLOVÁ, J. *Operační výzkum*. 2. vyd. OSTRAVA: Vysoká škola báňská, 2002. 217 s. ISBN 80-248-0190-6

[6] TUZAR, A., MAXA, P., SVOBODA, V. *Teorie dopravy*. 1. vyd. PRAHA: České vysoké učení technické, 1997. 278 s. ISBN 80-01-01637-4

[7] VEJMOLA, S., FIALA, P. *Teorie grafů*. 1. vyd. PRAHA: Státní pedagogické nakladatelství, 1985. 222 s. ISBN

[8] VOLEK, J. *Operační výzkum I.* 1. vyd. PARDUBICE: Univerzita Pardubice, 2002. 111 s. ISBN 80-7194-410-6

elektronické zdroje

[9] BERKA, M. *Operační výzkum* [online]. BRNO: Vysoké učení technické v Brně, 1999 [cit. 2008-04-14]. Dostupný z WWW: < http://home.eunet.cz/berka/o/ >

[10] HLINĚNÝ, P. *Optimalizační úlohy* [online]. BRNO: Masarykova univerzita, 2007 [cit. 2008-04-01]. Dostupný z WWW: < http://www.fi.muni.cz/~hlineny/Teaching/OU/OU-text07.pdf >

[11] JANČAR, P. *Vyčíslitelnost a složitost* [online]. OSTRAVA: Vysoká škola báňská, 2005 [cit. 2008-03-24]. Dostupný z WWW: < http://www.cs.vsb.cz/jancar/TEORET-INF/VAS-PREDNASKY/vas-predn-06-ho.pdf >

[12] PATEL, A. *Participation and Cooperation in Action* [online]. WASHINGTON: An Official Informs publication, 2003 [cit. 2008-03-23]. Dostupný z WWW: < http://ormstomorrow.informs.org/archive/spring03/spotlight.htm >

Seznam zkratek a symbolů

HK	Hamiltonova kružnice
KM	Kimova metoda
LA	Littleův algoritmus
MS	Microsoft systems
NDNV	metoda nejbližšího dosud nenavštíveného vrcholu
OV	operační výzkum
POC	problém obchodního cestujícího
RS	rozhodovací strom
ú.f.	účelová funkce
VBA	Visual Basic for Applications

Prohlášení o využití výsledků diplomové (bakalářské) práce

Prohlašuji, že

- byl(a) jsem seznámen(a) s tím, že na mou diplomovou (bakalářskou) práci se plně vztahuje zákon č. 121/2000 Sb. – autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo,
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně ke své vnitřní potřebě diplomovou (bakalářskou) práci užít (§ 35 odst. 3),
- souhlasím s tím, že jeden výtisk diplomové (bakalářské) práce bude uložen v Ústřední knihovně VŠB-TUO k prezenčnímu nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové (bakalářské) práce. Souhlasím s tím, že údaje o diplomové (bakalářské) práci, obsažené v Záznamu o závěrečné práci, umístěném v příloze mé diplomové (bakalářské) práce, budou zveřejněny v informačním systému VŠB-TUO,
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona,
- bylo sjednáno, že užít své dílo – diplomovou (bakalářskou) práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).

V Ostravě dne ………………………..

……………………………………………..
jméno a příjmení studenta

Adresa trvalého pobytu studenta:

…………………………………………………………

Příloha č. 1: Matematický zápis problému

Matematický zápis problému

Jako úlohu lineární programování můžeme problém prvního okruhu zapsat podle vzorců 2.1 až 2.5 následujícím způsobem.

$z = 0 * x_{11} + 8980 * x_{12} + 8560 * x_{13} + 8370 * x_{14} + 8980 * x_{15} + 9190 * x_{16} + 9760 * x_{17} +$
$10020 * x_{18} + 9250 * x_{19} + 9570 * x_{110} + 10960 * x_{111} +$
$8980 * x_{21} + 0 * x_{22} + 2310 * x_{23} + 2900 * x_{24} + 2720 * x_{25} + 3460 * x_{26} + 4030 * x_{27} +$
$4370 * x_{28} + 3610 * x_{29} + 3660 * x_{210} + 3620 * x_{211} +$
$8560 * x_{31} + 2090 * x_{32} + 0 * x_{33} + 740 * x_{34} + 1240 * x_{35} + 1800 * x_{36} + 2100 * x_{37} +$
$2350 * x_{38} + 1580 * x_{39} + 1900 * x_{310} + 3350 * x_{311} +$
$8370 * x_{41} + 2440 * x_{42} + 260 * x_{43} + 0 * x_{44} + 1050 * x_{45} + 1610 * x_{46} + 1930 * x_{47} +$
$2200 * x_{48} + 1440 * x_{49} + 17400 * x_{410} + 3800 * x_{411} +$
$9120 * x_{51} + 2650 * x_{52} + 1240 * x_{53} + 550 * x_{54} + 0 * x_{55} + 1880 * x_{56} + 2330 * x_{57} +$
$2610 * x_{58} + 1840 * x_{59} + 1850 * x_{510} + 3610 * x_{511} +$
$9190 * x_{61} + 3540 * x_{62} + 2190 * x_{63} + 2030 * x_{64} + 2230 * x_{65} + 0 * x_{66} + 2520 * x_{67} +$
$2760 * x_{68} + 1990 * x_{69} + 2310 * x_{610} + 5140 * x_{611} +$
$9760 * x_{71} + 4120 * x_{72} + 2090 * x_{73} + 1930 * x_{74} + 2330 * x_{75} + 2520 * x_{76} + 0 * x_{77} +$
$870 * x_{78} + 530 * x_{79} + 1050 * x_{710} + 4970 * x_{711} +$
$10020 * x_{81} + 4370 * x_{82} + 2360 * x_{83} + 2200 * x_{84} + 2580 * x_{85} + 2770 * x_{86} + 870 * x_{87} +$
$0 * x_{88} + 770 * x_{89} + 1300 * x_{810} + 519 * x_{811} +$
$9250 * x_{91} + 3590 * x_{92} + 1590 * x_{93} + 1440 * x_{94} + 1810 * x_{95} + 2000 * x_{96} + 530 * x_{97} +$
$770 * x_{98} + 0 * x_{99} + 850 * x_{910} + 4560 * x_{911} +$
$9870 * x_{101} + 3640 * x_{10}2 + 1990 * x_{103} + 1730 * x_{104} + 1700 * x_{105} + 22320 * x_{106} +$
$1050 * x_{107} + 1280 * x_{108} + 640 * x_{109} + 0 * x_{1010} + 3600 * x_{1011} +$
$10960 * x_{111} + 293 * x_{112} + 3340 * x_{113} + 3830 * x114 + 3610 * x_{115} + 5140 * x_{116} +$
$4970 * x_{117} + 5190 * x_{118} + 4560 * x_{119} + 3900 * x_{1110} + 0 * x_{1111}$

Účelovou funkci z budeme minimalizovat, protože nás zajímá nejkratší možné spojení přes uzly u_1 až u_{11}.

za podmínek:

$x_{11} + x_{21} + x_{31} + x_{41} + x_{51} + x_{61} + x_{71} + x_{81} + x_{91} + x_{101} + x_{111} = 1$

$x_{12} + x_{22} + x_{32} + x_{42} + x_{52} + x_{62} + x_{72} + x_{82} + x_{92} + x_{102} + x_{112} = 1$

$x_{13} + x_{23} + x_{33} + x_{43} + x_{53} + x_{63} + x_{73} + x_{83} + x_{93} + x_{103} + x_{113} = 1$

$x_{14} + x_{24} + x_{34} + x_{44} + x_{54} + x_{64} + x_{74} + x_{84} + x_{94} + x_{104} + x_{114} = 1$

$x_{15} + x_{25} + x_{35} + x_{45} + x_{55} + x_{65} + x_{75} + x_{85} + x_{95} + x_{105} + x_{115} = 1$

$x_{16} + x_{26} + x_{36} + x_{46} + x_{56} + x_{66} + x_{76} + x_{86} + x_{96} + x_{106} + x_{116} = 1$

$x_{17} + x_{27} + x_{37} + x_{47} + x_{57} + x_{67} + x_{77} + x_{87} + x_{97} + x_{107} + x_{117} = 1$

$x_{18} + x_{28} + x_{38} + x_{48} + x_{58} + x_{68} + x_{78} + x_{88} + x_{98} + x_{108} + x_{118} = 1$

$x_{19} + x_{29} + x_{39} + x_{49} + x_{59} + x_{69} + x_{79} + x_{89} + x_{99} + x_{109} + x_{119} = 1$

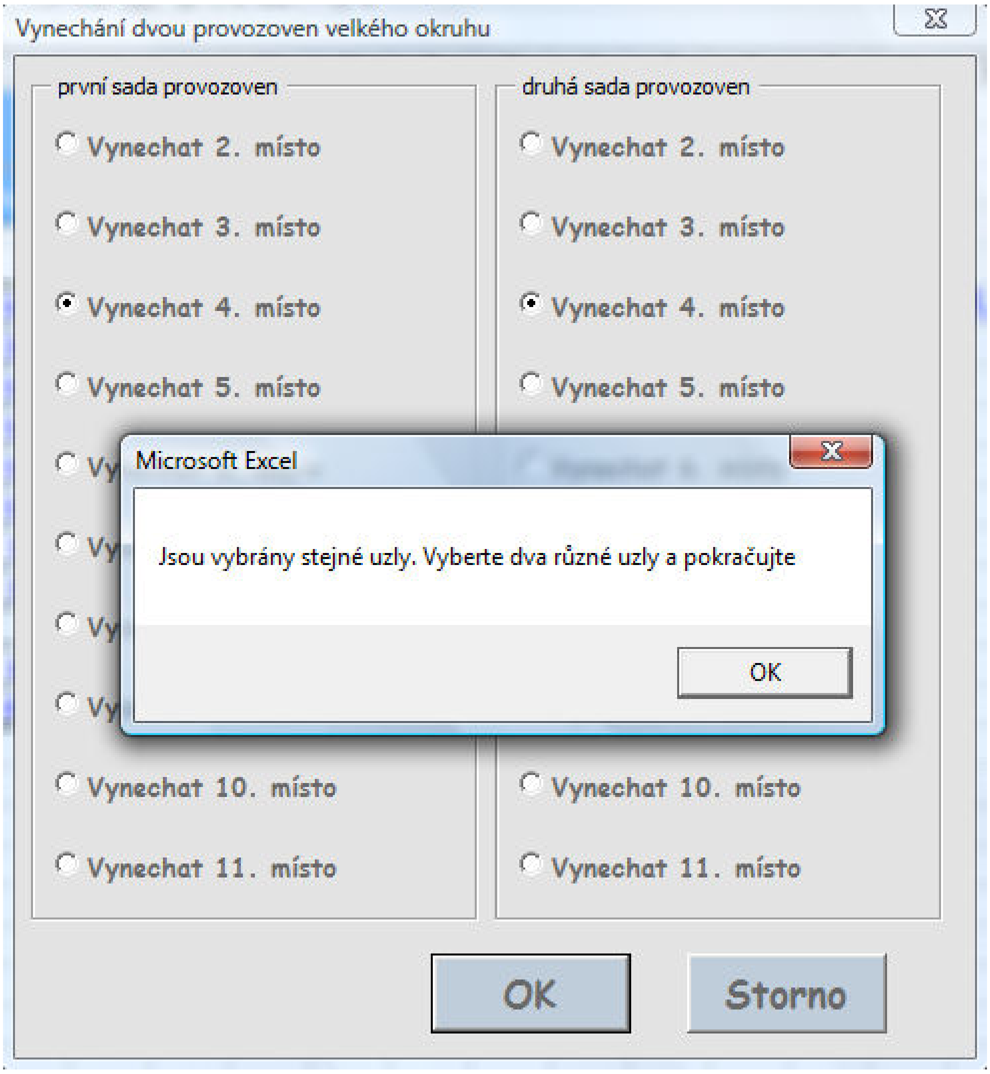
$x_{110} + x_{210} + x_{310} + x_{410} + x_{510} + x_{610} + x_{710} + x_{810} + x_{910} + x_{1010} + x_{1110} = 1$

$x_{111} + x_{211} + x_{311} + x_{411} + x_{511} + x_{611} + x_{711} + x_{811} + x_{911} + x_{1011} + x_{1111} = 1$

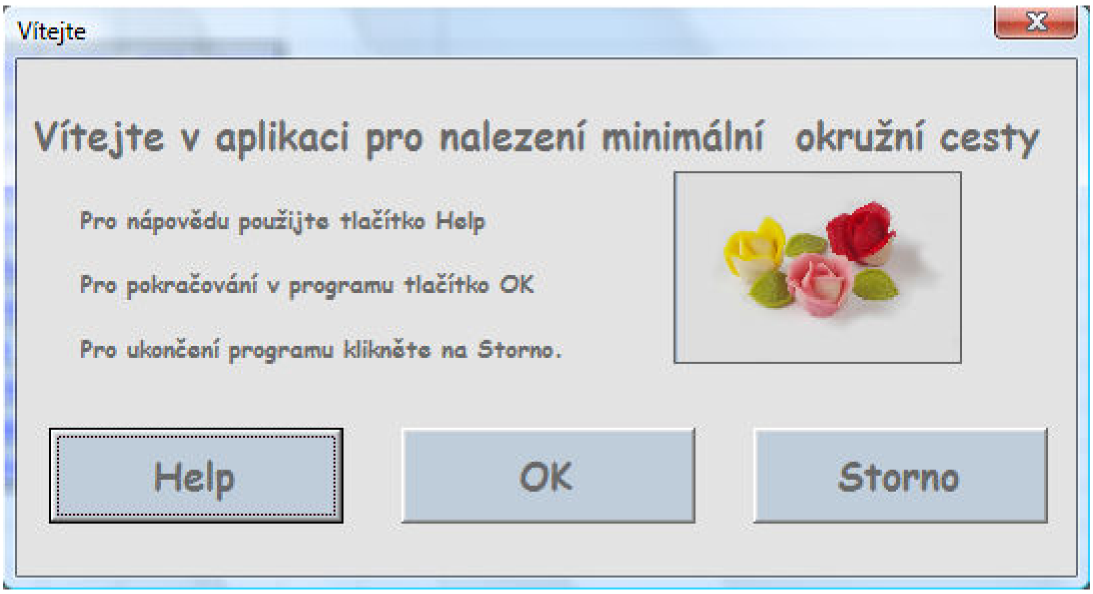
$x_{ij} = \{0;1\}$, bude – li například $x_{12} = 0$, úsek nebude zařazen do naší minimální Hamiltonovy kružnice.

Podmínky x_{11}, x_{22}, x_{33}, x_{44}, x_{55}, x_{66}, x_{77}, x_{88}, x_{99}, x_{1010}, x_{1111} jsou uvedeny jen pro úplnost řešení. Protože se však jedná o smyčky, které jsou v našem případě nulové délky, není třeba je brát do úvahy. Ve vypsané účelové funkci jsou tyto úseky vynulovány, násobíme je nulovou délkou.

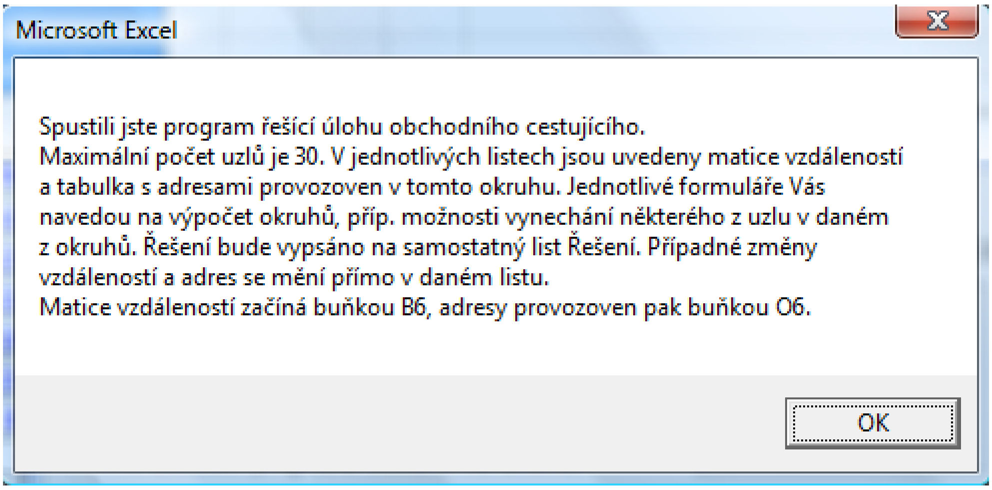
Příloha č. 2: Ukázka výstupů aplikace



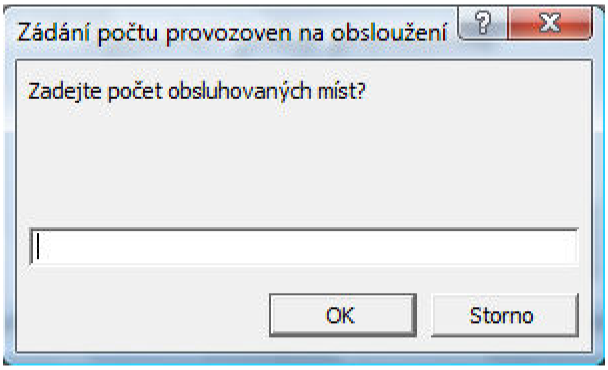
Obr. 4.6: Vynechání dvou provozoven + chybová hláška



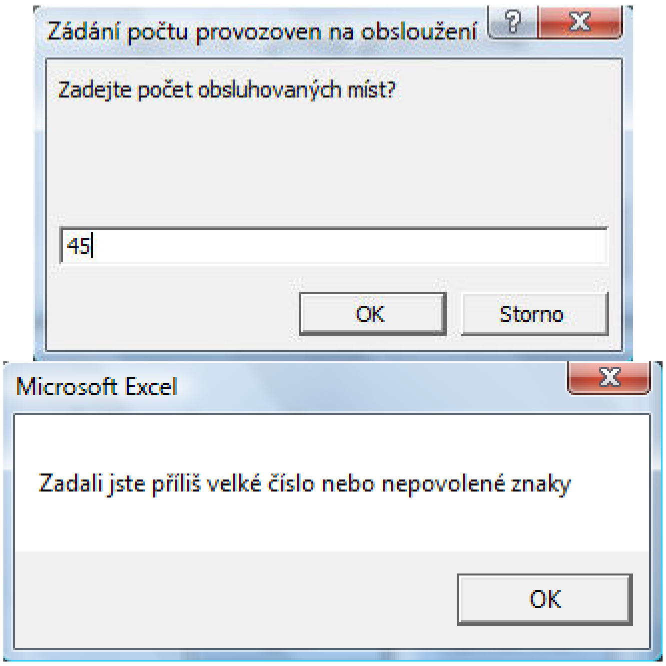
Obr. 4.7: Úvodní formulář



Obr. 4.8: Nápověda k programu



Obr. 4.9: Způsob zadávání počtu provozoven



Obr. 4.10: Nepovolený počet provozoven